THE ONTARIO WASTE MANAGEMENT SYSTEMS MODEL

STANDARDS DEVELOPMENT BRANCH MOE
3693600000427[illegible]

June 1980

Ministry
of the
Environment

Ontario

The Honourable
Harry C. Parrott, D.D.S.,
Minister

Graham W. S. Scott, Q.C.,
Deputy Minister

TD
897
.J46
O58
1980

Copyright Provisions and Restrictions on Copying:

This Ontario Ministry of the Environment work is protected by Crown copyright (unless otherwise indicated), which is held by the Queen's Printer for Ontario. It may be reproduced for non-commercial purposes if credit is given and Crown copyright is acknowledged.

It may not be reproduced, in all or in part, part, for any commercial purpose except under a licence from the Queen's Printer for Ontario.

For information on reproducing Government of Ontario works, please contact Service Ontario Publications at copyright@ontario.ca

THE ONTARIO WASTE MANAGEMENT SYSTEMS MODEL

Developed for the Ontario Ministry of the Environment

Waste Management Branch

Dr. L. Jenkins

June, 1980

TD
897
.J46
058
1980

ABSTRACT

This report documents a computer model developed to aid in designing a regional solid waste management system with the option of waste reclamation. Input to the model consists of data on the amount of waste generated at each source, location and size limits of possible transfer stations, incinerators, separation plants and landfills, cost of such facilities, and haulage costs between locations. After preliminary editing and calculations from the raw data, a mixed integer program selects those facilities and material flows which will minimize total system cost, net of any revenue from reclaimed materials. The resulting solution is printed out in summary and detailed form for examination. By making minor modifications to the input, the user can analyse a variety of different scenarios to aid in his final system selection.

ACKNOWLEDGEMENTS

A number of people have helped with the design, programming and debugging of this model. At the Ontario Ministry of the Environment, J. Burnham and A. Ciulini worked on the model specifications, aided by J. Polak who later took over maintenance of the model. B. Kirby of James F. MacLaren Ltd. helped in the initial testing and validation of the model. At the University of Windsor, D. McKenzie programmed most of the Report Writer Program and C. Lau helped with debugging much of the program coding.

MOE
STANDARDS DEVELOPMENT BRANCH
LIBRARY

TABLE OF CONTENTS

PAGE

ABSTRACT i
ACKNOWLEDGEMENTS ii
TABLE OF CONTENTS iii
LIST OF APPENDICES viii
LIST OF FIGURES ix

PART I - MODEL OVERVIEW

1. INTRODUCTION 1
   1.1 Report Format 1
   1.2 The Model 1
   1.3 Computer Requirements 2

2. MODEL DESCRIPTION 5
   2.1 The Edit and Setup Program 5
   2.2 The Mixed Integer Program 7
   2.3 The Report Writer Program 7

PART II - USER'S GUIDE

3. THE MODELLING APPROACH 9
   3.1 A One-Period Model 9
   3.2 Modelling Source Generation 9
   3.3 Modelling Transportation Costs 9
   3.4 Modelling Facility Operating Costs 10
   3.5 Modelling Facility Capital Costs 10
   3.6 Modelling Markets 11
   3.7 The Mixed Integer Formulation 11

3.8 Pegging Facilities 12
3.9 Computer Costs 12

4. DATA INPUT FILES 13
4.1 Numbering Conventions for Locations in the Model 13
4.2 Data for the Edit and Setup Program 14
4.2.1 General Parameter Subfile 16
4.2.2 Vehicle Subfile 18
4.2.3 Source Subfile 19
4.2.4 Transfer Station Subfile 22
4.2.5 Plant Subfile 25
4.2.6 Landfill Subfile 27
4.2.7 Market Subfile 29
4.2.8 Transportation Link Subfile 30
4.3 Error Messages from the Edit and Setup Program 32
4.4 Input for the Report Writer Program 34

5. STEPS IN RUNNING THE MODEL 36
5.1 Checking the Data with the Edit and Setup Program 36
5.2 First Run of the Mixed Integer Program 37
5.3 First Run of the Report Writer Program 37
5.4 "Production" Runs of the Model 37

PART III - PROGRAMMER'S GUIDE

6. RUNNING THE MODEL ON A CDC COMPUTER 39
6.1 Setting Device Numbers, Date and Dimension Statements in the ESP 39
6.2 Setting Device Numbers, Lines Per Page, Date and Dimension Statements in the RWP 41
6.3 JCL for Running on CDC with APEX-III 42
6.4 The Control Program for APEX-III 42

7. RUNNING THE MODEL ON AN IBM COMPUTER 46
7.1 Setting Device Number, Date and Dimension Statements in the ESP 46

7.2 Setting Device Numbers, Lines Per Page, Date and Dimension Statements in the RWP 47
7.3 JCL for Running on IBM/370 with MPSX-MIP 49
7.4 The Control Program for MPSX-MIP 49

8. DOCUMENTATION OF THE EDIT AND SETUP PROGRAM 52
8.1 Main Routine 52
8.1.1 Glossary of Important Variables 52
8.2 Subroutine START 55
8.2.1 Glossary of Important Variables 55
8.2.2 Processing 55
8.3 Subroutine SOURCE 55
8.3.1 Glossary of Important Variables 55
8.3.2 Calculations 56
8.4 Subroutine TRANFR 58
8.4.1 Glossary of Important Variables 58
8.4.2 Calculations 59
8.5 Subroutine PLANTS 60
8.5.1 Glossary of Important Variables 60
8.5.2 Calculations 61
8.6 Subroutine LANDFL 62
8.6.1 Glossary of Important Variables 62
8.6.2 Calculations 62
8.7 Subroutine MARKET 63
8.7.1 Glossary of Important Variables 63
8.8 Subroutine LINKEM 64
8.8.1 Glossary of Important Variables 64
8.8.2 Procedure 65
8.9 Subroutine TSETUP 65
8.9.1 Glossary of Important Variables 65
8.9.2 Procedure 66
8.10 Subroutine DSETUP 66

8.11 Subroutine REGRESS 66
8.11.2 Glossary of Important Variables 67
8.11.2 Calculations 67
8.12 Function USPWF 68
8.12.1 Glossary of Important Variables 68
8.12.2 Calculations 68

9. DOCUMENTATION OF THE REPORT WRITER PROGRAM 69
9.1 Main Program 69
9.2 Subroutine DFETCH 69
9.3 Subroutine TFETCH for MPSX-MIP 69
9.3.1 Glossary of Important Variables 69
9.3.2 Procedure 71
9.4 Subroutine TFETCH for APEX-III 71
9.4.1 Glossary of Important Variables 71
9.4.2 Procedure 72
9.5 Subroutine EXSUM 72
9.5.1 Glossary of Important Variables 72
9.5.2 Procedure 73
9.6 Subroutine SUMREP 73
9.6.1 Glossary of Important Variables 74
9.6.2 Procedure 74
9.7 Subroutine DETGEN 74
9.7.1 Glossary of Important Variables 74
9.7.2 Procedure 75
9.8 Subroutine DETFAC 75
9.8.1 Glossary of Important Variables 75
9.8.2 Procedure 76
9.9 Subroutine MKTREP 76
9.9.1 Glossary of Important Variables 76
9.9.2 Procedure 76

9.10 Subroutine TPSHIP 76
    9.10.1 Glossary of Important Variables 77
    9.10.2 Procedure 77

REFERENCES
APPENDICES

LIST OF APPENDICES

1. Mathematical Formulation of the Model
2. Sample Data Input File
3. Output From ESP
4. Intermediate TFILE
5. Output from APEX-III
6. Output from RWP
7. Source Code of ESP for CDC
8. Source Code of RWP for CDC
9. Source Code of TFETCH for IBM

LIST OF FIGURES

PAGE

1. Typical Components in a Regional Solid Waste Management System 3

2. A Computer Generated Solid Waste Management System 4

3. Schematic of the Modelling System 6

4. Organization of Data File for the Edit and Setup Program 15

5. Organization of Transportation Link Subfile 31

6. File Names Used in APEX-III Version 40

7. JCL for Running on CDC Under SCOPE 43

8. JCL for Running on CDC Under NOS 44

9. Control Program for APEX-III 45

10. File Names Used in MPSX-MIP Version 48

11. JCL for Running on IBM/370 Under OS 50

12. Control Program for MPSX-MIP 51

13. Sequence for Execution of Subroutines in ESP 53

14. Common Blocks Needed in ESP 54

15. Sequence for Execution of Subroutines in RWP 70

PART I

MODEL OVERVIEW

CHAPTER ONE

INTRODUCTION

1.1 Report Format

The report is divided into three parts and an appendix. The first part describes the main features and applicability of the model. The second part, the User's Guide, describes the data needed by the model and the format in which it should be prepared. Error messages from the Edit and Setup Program are listed and the meaning of the model output is discussed. The steps which should be followed in the early stages of using the model for a regional study are also outlined. The third part, the Programmer's Guide, is made up of a chapter on how to set up and run the model on a CDC computer, a chapter on how to set up and run the model on an IBM computer, and detailed documentation of the Edit and Setup Program (ESP) and the Report Writer Program (RWP). The appendices include the mathematical formulation of the model, sample output reports, and listings of the ESP and RWP.

The report makes no attempt to compare the model with other models available for a similar purpose. For a detailed review of other models the reader should see reports (7,8) cited in the References.

1.2. The Model

The model is intended to help a decision-maker in choosing the best solid waste management system (SWMS) to implement in a particular region over some specified time horizon. The analysis starts with the following information:

a) a given planning horizon for the regional study

b) a number of sources of waste (unit generating areas) and their predicted waste generation over the planning horizon

c) possibly a number of existing facilities (transfer stations, incinerators, landfills) and their operating costs and throughput limits

d) a number of potential facilities (transfer stations, reclamation plants, landfills) and their capital and operating costs and throughput limits

e) if waste reclamation is an option, details for each market location of price payable and the upper limit on the amount of material the market location will purchase

f) transport costs along all links which might be used in the SWMS.

Given all the above information, the model will select the SWMS which disposes of all the waste and satisfies all constraints at least net cost. Figure 1 illustrates a typical region with a number of sources from which the waste must be removed and potential processing and disposal facilities. Figure 2 illustrates the facilities and material shipments selected by the model.

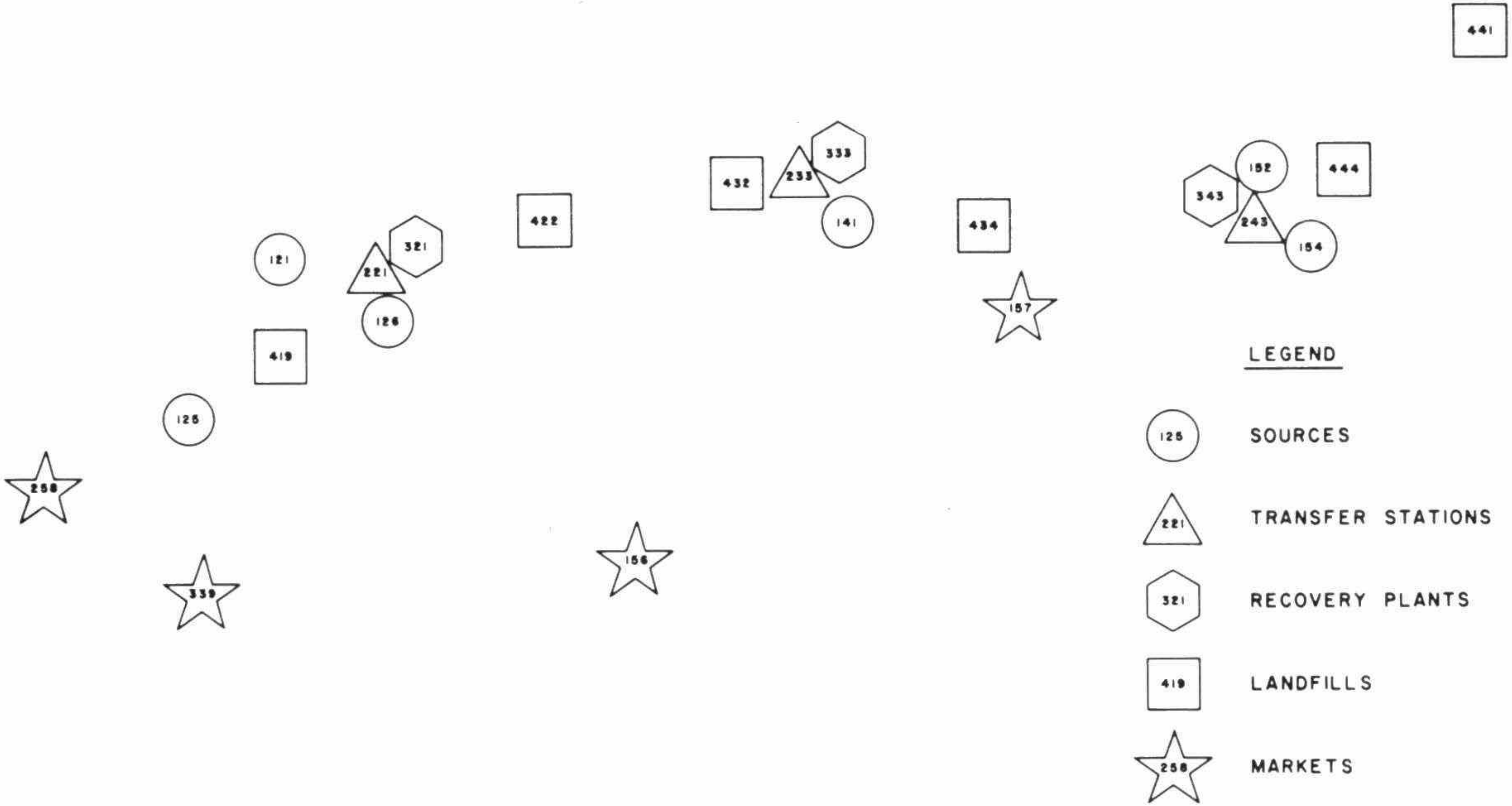
In practice the decision-maker may not be sure of all the data in the above list or might wish to examine the effect of making different facilities available or tightening various constraints. Only minor modifications of the input file are needed to change critical data. With such changes the user of the model can examine the results for a variety of different scenarios.

There are two outputs from the model. The first lists in detail the data entered and some of the intermediate calculated values. This allows the user to check that the data are as intended, and provides a listing for later reference. The second lists the solution found by the model, including system cost, facilities used, facility costs and throughput, material flows and transportation costs.

1.3 Computer Requirements

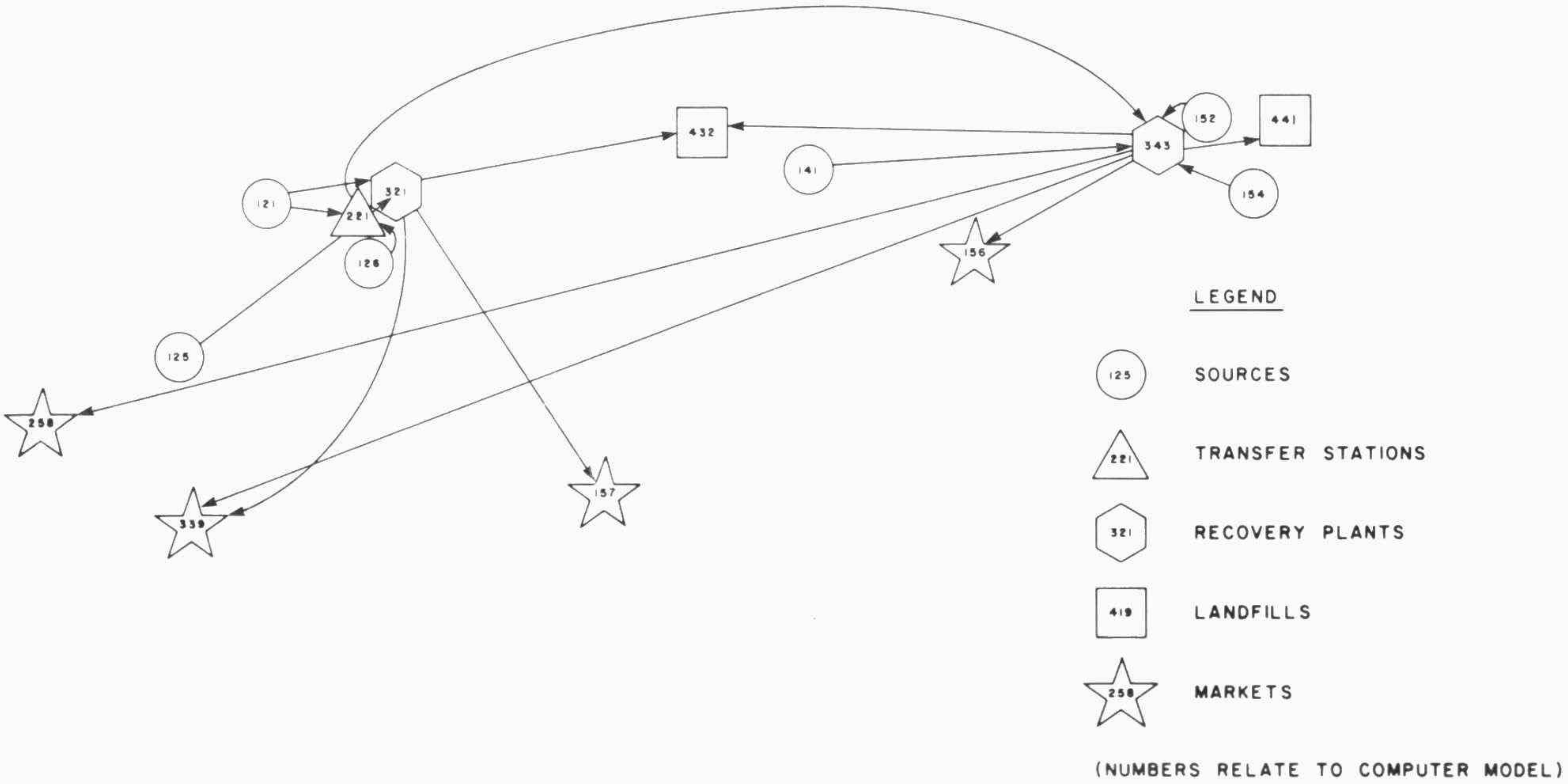
The model has been designed to run on a larger computer of Control Data Corporation (CDC) or International Business Machines (IBM). The CDC version of the model uses the package program APEX-III with the mixed integer module while the IBM version uses the package MPSX with the mixed integer module MIP. Any CDC computer large enough to run APEX-III with the mixed integer module, or any IBM computer which can run MPSX-MIP, can run the rest of the model which is programmed in FORTRAN IV.

FIGURE 1 - TYPICAL COMPONENTS IN A REGIONAL SOLID WASTE MANAGEMENT SYSTEM



(NUMBERS RELATE TO COMPUTER MODEL)

FIGURE 2 - A COMPUTER GENERATED SOLID WASTE MANAGEMENT SYSTEM



CHAPTER TWO

MODEL DESCRIPTION

This chapter describes the model without going into great details about data preparation, which is described in Part II or specific programming details, which are given in Part III.

Figure 3 is a system flowchart of the model. The model is made up of three programs, the Edit and Setup Program (ESP), the Mixed Integer Program (MIP) and the Report Writer Program (RWP). The programs must be run in this sequence, though intermediate files may be saved until the user is satisfied that the data are correct and a program has performed satisfactorily before running the next program.

2.1 The Edit and Setup Program

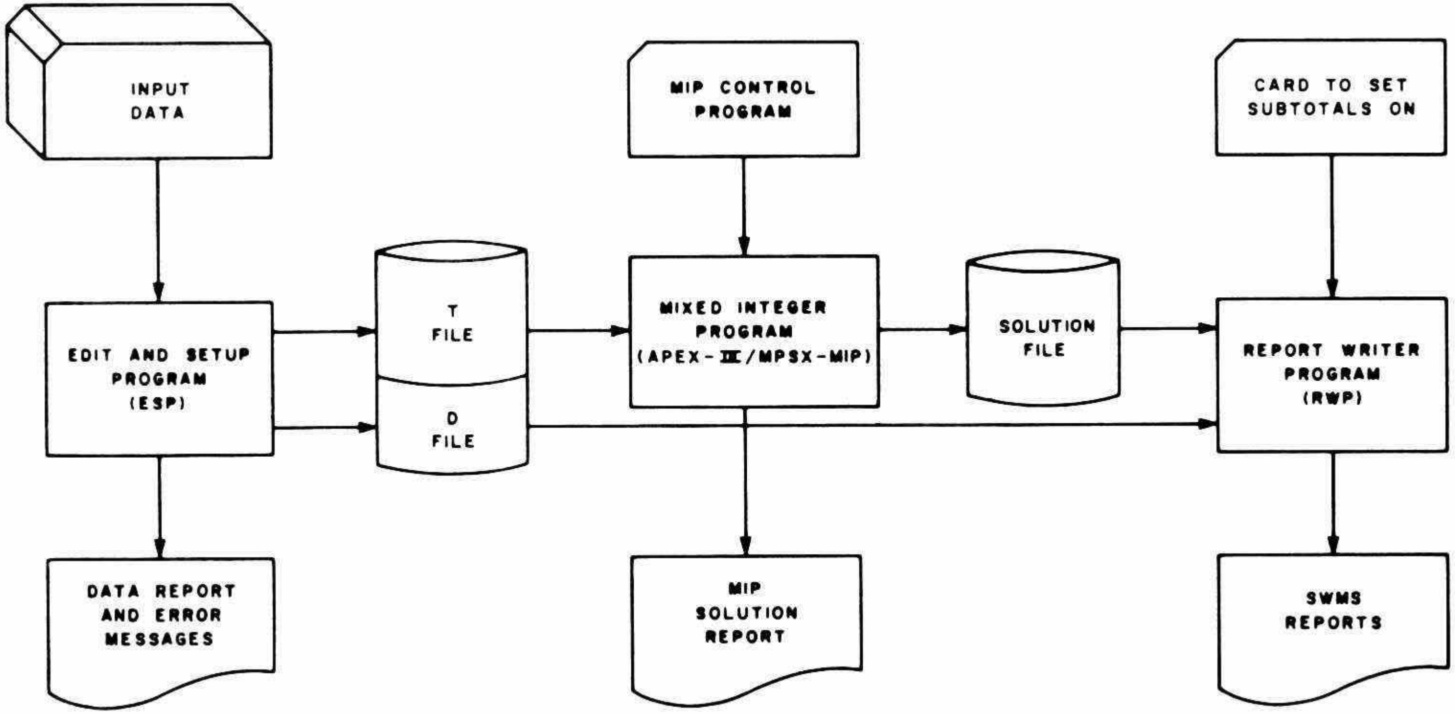
The ESP takes raw data in a form convenient to the user and, after checking for inconsistencies, prepares files to be used by the MIP and RWP. The user has the option of listing all input and all calculated values, only part of the input and calculated values, or only error messages.

The first few cards read in by the program are titles and basic information such as discount rate to be used on cash flows, number of sources, possible transfer stations, plants, landfills, vehicle names and vehicle operating costs.

For each source the total waste quantity generated over the study period must be given. The user can do this in several formats involving total annual generation, per capita generation rates, population and population growth rates.

For each possible transfer station, plant and landfill, the user must provide capital and operating costs and an upper limit on facility capacity. For plants, the proportion of waste lost in processing or recovered as various resources must be specified. The user can specify whether each facility must be available in the computer run, should be excluded from the run, or whether the MIP should decide to use or not use the facility. If the

FIGURE 3 - SCHEMATIC OF THE MODELLING SYSTEM


INPUT DATA
MIP CONTROL PROGRAM
CARD TO SET SUBTOTALS ON
EDIT AND SETUP PROGRAM (ESP)
T FILE
D FILE
MIXED INTEGER PROGRAM (APEX-III/MPSX-MIP)
SOLUTION FILE
REPORT WRITER PROGRAM (RWP)
DATA REPORT AND ERROR MESSAGES
MIP SOLUTION REPORT
SWMS REPORTS

analysis allows for resource recovery, data are required on material price and demand at the various market locations.

The final part of the raw data details every transportation link in the model along which material might flow. The origin, destination, travel time and vehicle to be used on the link are specified.

2.2 The Mixed Integer Program

The MIP package used is either APEX-III (1) if the model is run on a CDC computer, or MPSX-MIP (5,6) if the model is run on an IBM computer. Both packages search for the minimum cost SWMS using the Simplex and Branch-and-Bound algorithms (4,9). While each package uses the same data, similar algorithms to search for the minimum cost SWMS, and produce the same result, the format of their input and output files is slightly different. This, together with the slight difference in CDC and IBM FORTRAN is the reason for two versions of the model.

2.3 The Report Writer Program

The RWP uses the output from the MIP and a file of location names passed from the ESP to produce a number of reports on the best system found by the MIP. The reports produced are:

1. Executive Summary - this summarizes system capital and operating cost, facilities used, markets supplied, and a cost breakdown by type of facility.
2. Waste Generation Area Shipments - this lists the shipments of waste from each source, the cost of shipping and their destinations.
3. Waste Generation Area Shipments Cost Summary - this lists, for each source, total and weighted average cost of transporting the waste from the source to its destination (s).
4. Facility Shipments - this lists the quantity of material shipped from each intermediate facility (transfer station or plant) together with the cost of shipping and destination.
5. Detailed Report of Facility Operations - this lists for each transfer station, plant or landfill, the total annual and per tonne capital, operating and transportation costs (in the case of an intermediate facility).

6. Marketing Report - this lists for each market location, by product, the total amount of resource received and total revenue obtained.

PART II

USER'S GUIDE

CHAPTER THREE

THE MODELLING APPROACH

This chapter describes the approaches taken in modelling the major components of a solid waste management system (SWMS) and how the mathematical formulation and computer model are derived. For further information the user should read the detailed description of the Edit and Setup Program (ESP) in Chapter Seven, and consult a standard Operations Research text such as (4, 9) for explanation of the mixed integer program formulation and solution procedure. This chapter aims only to explain the modelling sufficiently that the user understands how to prepare the data and read the model output.

3.1 A One-Period Model

It is important to remember in all that follows that the Ontario Waste Management Model is a one-period model. This means that the facilities the model chooses to use, the size and throughput of these facilities, and the pattern of material shipments do not change within the period modelled. This does not preclude the user from dividing the total study horizon into several periods and anlysing each separately. It means only that the model does not deal with several periods simultaneously.

3.2 Modelling Source Generation

The first information required by the model is an estimate for waste quantity generated. It is assumed that the study region is divided into unit generating areas which, for the purposes of the model, can be treated as point sources. The model needs data on the amount of waste generated at each source over the whole period. The amount may be given in a variety of formats depending on the availability of data.

3.3 Modelling Transportation Costs

Two factors are used to calculate the cost of moving a tonne (metric ton) of material along a transportation link. The first is the haulage cost per tonne per hour on the vehicle which will be used. The second is the round-trip travel time on the link. Up to eight different kinds of vehicles can be used in the model. Their costs are entered in the vehicle subfile. In the transportation link subfile, for each link the origin, destination, type of vehicle to be used, (effectively giving the cost per tonne per hour) and the round trip travel time are given.

Since transportation costs are incurred in every year of the period, they are discounted back to day one of the first year.

3.4 Modelling Facility Operating Costs

The throughput of a facility (transfer station, incinerator, separation plant, landfill) is decided by the model and can take any value between the lower and upper limits specified by the user. Thus the model needs an operating cost curve, rather than just the cost for a single throughput. The user provides operating cost figures for a number of different throughputs of the same facility and the model calculates linear regression coefficients using the method of least squares. The 'fixed' cost is assumed to be incurred if the facility has any throughput above zero. If just one throughput cost figure is entered, then a fixed cost will be used for any throughput greater than zero.

Since operating costs are incurred over each year of the period they are discounted back to day one of the first year.

3.5 Modelling Facility Capital Costs

Before a facility can be operated, it must be built or developed. If it already exists, it may be available at zero capital cost. Then a single capital cost figure of zero can be put in the model. Improving or expanding an existing facility, or developing a new one, incurs a capital cost which the model assumes to be spent on day one of the first year. However, the user is given the option of making the payment either as an amortization stream or as one or more lump payments. Amortization payments or lump payments are discounted by the model to day one of the first year.

The capacity of the facility will be the same as the throughput which is decided by the model (an exception is an existing facility for which the capital cost figure is zero; then the existing capacity is the figure entered for limit on throughput). The model derives a capital cost curve by linear regression on two or more different sizes of the facility. If cost information is given for just one size of facility, then this cost is assumed for any size greater than zero.

3.6 Modelling Markets

Markets are classified by resource recovered and by location within each resource. For each market location the user provides the annual demand and the price payable per tonne. Shipments will result in a steady stream of revenue, so this is discounted to day one of year one of the period.

3.7 The Mixed Integer Formulation

Finding the 'best' SWMS comes down to deciding which of the possible facilities should be used, their throughputs and what material should be shipped where. 'Best' is taken to mean minimum total discounted cost. This is made up of facility capital costs, facility operating costs, transportation costs, less any revenue from sales of recovered materials.

Any system must satisfy a large number of constraints. Obviously all waste must be removed from sources and all material entering an intermediate facility must leave it or be lost in processing. Upper limits apply to facility throughput and shipments to market locations.

Some constraints which are included in the formulation and may be set by the user to explore different scenarios or environmental policies are

i) limit on total tonnes of waste landfilled (this can be used to force incineration or reclamation)
ii) limit on capital expenditure
iii) limit on the number of transfer stations
iv) limit on the number of plants (incinerators and reclamation plants)
v) limit on the number of landfills.

Because facility costs are modelled as a fixed cost plus a variable cost proportional to throughput, and transportation costs and revenue are directly proportional to quantity, the problem of finding the lowest cost SWMS can be modelled as a mixed integer program (MIP) (See (4) or (9)). Finding the lowest cost SWMS that satisfies all constraints, is performed by either the Control Data Corporation package APEX-III (1) or the International Business Machine package MPSX-MIP (5,6).

3.8 Pegging Facilities

As stated above, the MIP solution package decides which facilities should be used or not used, their related throughputs and material flows. However, in exploring different options the user may wish to 'limit the choice' of the MIP and specify in one computer run that certain facilities must not be used, or others must be used. The ability to do this is built into the model. The user simply specifies a parameter which defines the status of a facility as 'not used', 'used' or 'model decides'. Specifying the status as 'not used' or 'used' will be referred to as pegging. In a run of the model, none, some, or all of the facilities may be pegged.

If a facility is pegged as 'not used' the meaning is obvious: the facility will not form part of the SWMS formed by the model, will have no throughput, and will incur no cost. If a facility is pegged as 'used' the effect is a little less obvious: the model is forced to incur the fixed capital and operating costs of the facility whether or not it has any throughput. Effectively the user forces the model to commit the facility fixed costs (as derived in the capital and operating facility cost regressions) and after that lets the model decide throughput. Note, however, the user may also force throughput by specifying a lower limit on throughput greater than zero.

3.9 Computer Costs

This chapter ends with some points about computer costs of running the model. Costs for running the Edit and Setup Program and Report Writer Program will be minor but it is possible for a single run of the MIP on a large data base to cost a few hundred dollars.

If the status of a facility is set as 'model decides' then this gives rise to an integer decision variable. The computer cost of running the MIP for a particular data base depends on the number of integer decision variables in the model. The more facilities with status 'model decides', the greater the computer cost of running the MIP. The more facilities pegged as 'not used' or 'used', the lower the cost of running the model. It is recommended the user set the status of a facility as 'model decides' only when he wants to exercise that option rather than treating it as a default. In early runs of the model, when the user is still confirming the input data are correct, the most appropriate default status for a facility is 'used'. The topic is discussed further in Chapter 4.

CHAPTER FOUR

DATA INPUT FILES

This chapter describes in detail how to prepare data for input to the Edit and Setup Program and the Report Writer Program.

4.1 Numbering Conventions For Locations In The Model

The following rules apply to allocating numeric labels for sources and facilities:

1. The numbers may range between 100 and 999
2. Using the same number for more than one location of the same type will cause errors. The same number may be used for different types of locations. For example, having two sources, both labelled 413, will cause errors, but it is acceptable to have a source numbered 413, a plant numbered 413 and a landfill numbered 413.
3. Model output follows the sequence used in entering data to the model and is independent of location number. For example, if data for source 413 is entered before data for source 217, then the output will report shipments from source 413 before reporting shipments from source 217.

The numeric label for market location is a three digit number. The first digit is the resource numbers 1 to 9. The last two digits are an arbitrary label for location. Resources recovered are automatically numbered 1 to 9. The first resource named is taken as resource number 1, the second named is taken as resource 2 and so on. All information on markets for resource 1 must be entered before information on resource 2, and so on. For a particular resource, market locations may be given an arbitrary number in the range 10-99. Note that all three digits are required in identifying the destination end of a plant-to-market transportation link.

4.2 Data For The Edit And Setup Program

All the data defining the model for a particular regional study are entered in the Edit and Setup Program as 80 character records. The model is designed to use the following units: dollars, tonnes, kilometres, years, hours and minutes. Since the only unit conversion in the model is between hours and minutes, the model will work equally well with English units, but many of the report headings would then be misleading.

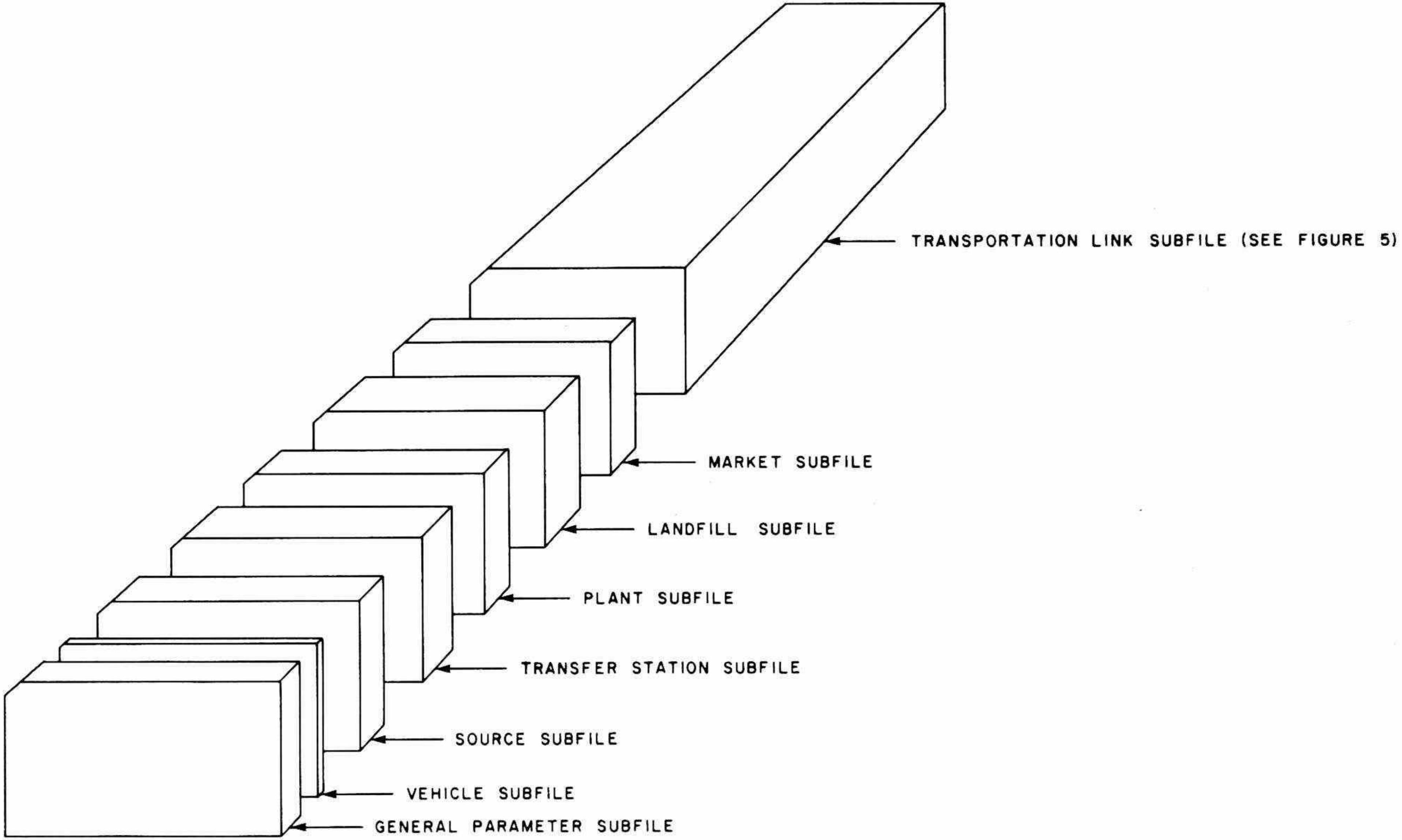
The data file is made up of eight subfiles, some of which may be empty if not applicable to the particular study. For example, if the analysis does not include resource recovery, then the market subfile will be empty.

The subfiles are:

A.	general parameter subfile	
B.	vehicle subfile	
C.	source subfile	
D.	transfer station subfile	may be empty
E.	plant subfile	may be empty
F.	landfill subfile	
G.	market subfile	may be empty
H.	transportation link subfile-	
	source to transfer	may be empty
	source to plant	may be empty
	source to landfill	may be empty
	transfer to plant	may be empty
	transfer to landfill	may be empty
	plant to landfill	may be empty
	plant to market	may be empty

The first seven subfiles are placed directly one behind the other. Each section of the transportation link subfile, including empty sections, must be ended with a card having a 9 in column 1 (Figure 4 and 5).

FIGURE 4 - ORGANISATION OF DATA FILE FOR THE EDIT AND SETUP PROGRAM



4.2.1 General Parameter Subfile (number of cards = 7)

This subfile, made up of 7 different kinds of cards gives general details about the model, including number of sources, facilities and other counts used in reading in the next six subfiles.

Card Type	Columns	Format	FORTRAN Name	Description
#1	1-40	A40	TITLE	Title of Study
#2	1-40	A40	RUNNAM	Description of this run
#3	2-5	I4	RUNBEG	First year of study period
	7-10	I4	RUNEND	Last year of study period
	11-20	F10.2	RATE	Discount rate (%)
	25	I1	INREP	Extent of output from ESP 0 - error messages only 1 - no listing of transport links 2 - complete report
	30	I1	OUTREP	Extent of output from RWP 0 - executive summary only 1 - complete report 2 - complete report and unformatted listing of data files (for error diagnosis)
	35	I1	TWRITE	Flag to write disk file for MIP 0 - do not write 1 - write file
	40	I1	DWRITE	Flag to write disk file for passing data from ESP to RWP 0 - do not write 1 - write file

Card Type	Columns	Format	FORTRAN Name	Description
	45	I1	LANDCD	Flag for linear approximation to landfill costs 0 - (fixed & variable) landfill costs 1 - linear approximation
#4	5	I1	NUMVEC	Number of vehicle types
	8-10	I3	NUMSRC	Number of sources
	14-15	I2	NUMTRF	Number of possible transfer stations
	19-20	I2	NUMPLT	Number of possible plants
	24-25	I2	NUMLFL	Number of possible landfills
	30	I1	NRESRC	Number of resources recoverable
#5	5	I1	NUMMKT(1)	Number of market locations for resource 1
	10	I1	NUMMKT(2)	Number of market locations for resource 2
	15	I1	NUMMKT(3)	Number of market locations for resource 3
	20	I1	NUMMKT(3)	Number of market locations for resource 4
	25	I1	NUMMKT(5)	Number of market locations for resource 5
	30	I1	NUMMKT(6)	Number of market locations for recource 6
	35	I1	NUMMKT(7)	Number of market locations for resource 7
	40	I1	NUMMKT(8)	Number of market locations for resource 8
	45	I1	NUMMKT(9)	Number of market locations for resource 9

Note: Card #5 will be blank if the number of resources recoverable NRESRC is 0.

Card Type	Columns	Format	FORTRAN Name	Description
#6	1-10	F10.0	LTOLFL	Limit to tonnes of waste landfilled for study region for whole period
	11-20	F10.0	LCAPTL	Limit to capital expenditure ($) for study region for whole period

Note: In card #6, if LTOLFL is blank or zero, the program sets it to 999,999,999. Similarly for LCAPTL

Card Type	Columns	Format	FORTRAN Name	Description
#7	1-10	F10.0	LNUMTR	Limit to number of transfer stations allowed in the run.
	11-20	F10.0	LNUMPL	Limit to number of plants allowed in the run
	21-30	F10.0	LNUMLF	Limit to number of landfills allowed in the run

4.2.2 Vehicle subfile (number of cards = 1 or 2)

This subfile gives the name and cost per tonne per hour for haulage vehicles. Data is read in for NUMVEC vehicles as given by card type 4 of the general parameter subfile.

Card Type	Columns	Format	FORTRAN Name	Description
#1	3-10	A8	VECNAM(1)	Name of vehicle type 1
	11-15	F5.3	COSTTH(1)	Cost per tonne per hour for vehicle 1
	18-25	A8	VECNAM(2)	Name of vehicle type 2

Card Type	Columns	Format	FORTRAN Name	Description
	26-30	F5.3	COSTTH(2)	Cost per tonne per hour for vehicle 2
	33-40	A8	VECNAM(3)	Name of vehicle type 3
	41-45	F5.3	COSTTH(3)	Cost per tonne per hour for vehicle 3
	48-55	A8	VECNAM(4)	Name of vehicle type 4
	56-60	F5.3	COSTTH(4)	Cost per tonne per hour for vehicle 4

Note: If required, a second card with the above format is used to enter VECNAM(5), COSTTH(5) to VECNAM(8), COSTTH(8).

4.2.3 Source Subfile (number of cards = 2 X number of sources)

This subfile gives a numeric label, a name and waste generation details for each source. For each source there are two cards, the first with identical format for each source, the second with format dependent on how waste generation data are entered. Data are read in for NUMSRC sources as given by card #4 of the general parameter subfile.

Card Type	Columns	Format	FORTRAN Name	Description
#0 (source header card)				
	1-3	I3	SRCIDN(J)	Numeric label for source
	4-11	A8	SRCNAM(J)	Name of source
	15	I1	IGENTP	Type of data for generating 1 - total waste over whole period 2 - total waste in first year with growth % 3 - total waste in different years (interpolate to total)

Card FORTRAN

Type	Columns	Format	Name	Description
				4 - tonnes per capita and population in first year, % growth
				5 - tonnes per capita and population in different years (interpolate to total)

The second card in the subfile is one of the cards #1 to #5 depending on the value of IGENTP

#1	(For IGENTP = 1)			
	1-3	I3	IDNUMB	Numeric label for source -must match SRCIDN(J) in preceding card
	11-20	F10.0	TAMOMT	Total waste generated over period (Tonnes)
#2	(For IGENTP=2)			
	1-3	I3	IDNUMB	Numeric label for source -must match SRCIDN(J) in preceding card
	11-20	F10.0	AMONT1	Total waste in first year
	26-30	F5.2	GROWTH	% growth in generation per year
#3	(For IGENTP = 3)			
	1-3	I3	IDNUMB	Numeric label for source-must match SRCIDN(J) in preceding card
	10	I1	NENTRY	Number of years for which data entered (2,3 or 4)
	12-15	I4	IYR(1)	First year for data entered (must match RUNBEG in card #3 of general parameter subfile)
	16-25	F10.0	AMOUNT(1)	Total waste in IYR(1)
	27-30	I4	IYR(2)	Second year for data entered

Card FORTRAN

Type	Columns	Format	Name	Description
	31-40	F10.0	AMOUNT(2)	Total waste in IYR(2)
	42-45	I4	IYR(3)	Third year for data entered (if NENTRY = 3 or 4)
	46-55	F10.0	AMOUNT(3)	Total waste in IYR(3)
	57-60	I4	IYR(4)	Fourth year for data entered (if NENTRY = 4)
	61-70	F10.0	AMOUNT(4)	Total waste in IYR(4)
Note:	Year of last entry must match RUNEND in card #3 of general parameter subfile			
#4	(For IGENTP = 4)			
	1-3	I3	IDNUMB	Numeric label for source-must match SRCIDN(J) in preceding card
	11-15	F5.3	TONYR1	Tonnes per capita in first year
	16-20	F5.2	TONGRO	% growth in generation per year
	21-30	F10.0	POPYR1	Population in first year
	31-35	F5.2	POPGRO	% growth in population per year
#5	(For IGENTP = 5)			
	1-3	I3	IDNUMB	Numeric label for source-must match SRCIDN(J) in preceding card
	10	I1	NENTRY	Number of years for which data entered (2,3 or 4)
	12-15	I4	IYR(1)	First year for data entered (must match RUNBEG in card #3 of general parameter subfile)
	16-20	F5.2	TONYRC(1)	Tonnes per capita in IYR(1)
	21-27	F7.0	POPYR(1)	Population in IYR(1)

Card FORTRAN

Type	Columns	Format	Name	Description
	29-32	I4	IYR(2)	Second year for data entered
	33-37	F5.2	TONYRC(2)	Tonnes per capita in IYR(2)
	38-44	F7.0	POPYR(2)	Population in IYR(2)
	46-49	I4	IYR(3)	Third year for data entered (if NENTRY = 3 or 4)
	50-54	F5.2	TONYRC(3)	Tonnes per capita in IYR(3)
	55-61	F7.0	POPYR(3)	Population in IYR(3)
	63-66	I4	IYR(4)	Forth year for data entered (if NENTRY = 4)
	67-71	F5.2	TONYRC(4)	Tonnes per capita in IYR(4)
	72-78	F7.0	POPYR(4)	Population in IYR(4)

Note: Year of last entry must match RUNEND in card #3 of general parameter subfile.

4.2.4 Transfer Station Subfile

This subfile gives details of each existing or potential transfer station. The header card gives a numeric label, name, an upper and lower limit on throughput (if used) and status of the station for this particular run of the model. By setting the status to 0 (not used) a station can be effectively excluded from a model run without any other changes to the data file.

Following each station header card are one or more cards each giving the capital cost which would be incurred if the station has a particular size. After this are one or more cards giving the annual operating cost which would be incurred if the station has a particular throughput. The number of these cards is given by the header card. Data is read in for NUMTRF transfer stations as given by card #4 of the general parameter subfile.

Card FORTRAN

Type	Columns	Format	Name	Description
#0 (transfer station header card)				
	1-3	I3	TRFIDN(J)	Numeric label for transfer station
	4-11	A8	TRFNAM(J)	Name of transfer station
	15	I1	TRSTAT(J)	Status in this run 0 - not used 1 - used 2 - model decides
	21-30	F10.0	TRFLLT(J)	Lower limit on annual throughput (if used)
	31-40	F10.0	TRFULT(J)	Upper limit on annual throughput (if used)
	45	I1	IWAY	Way of entering capital costs 0 - amortization 1 - lump payments
	50	I1	NCPENT	Number of different size stations for which capital data entered. NCPENT = 1, 2, 3, 4 or 5
	55	I1	NOPENT	Number of cost figures entered for different operating throughput. NOPENT = 1, 2, 3, 4 or 5
#1A (if IWAY = 0)				
	1-3	I3	IDNUM	Numeric label for transfer station - must match TRFIDN(J) in preceding header card
	11-20	F10.0	THRU	Annual capacity
	27-30	I4	ENDPAY	Last year of amortization payments
	31-40	F10.0	AMTPAY	Annual amortization payments

Card Type	Columns	Format	FORTRAN Name	Description
#1B (if IWAY = 1)				
	1-3	I3	IDNUM	Numeric label for transfer station - must match TRFIDN(J) in preceding header card
	10	I1	NLUMP	Numer of lump payments (1, 2 or 3)
	11-20	F10.0	THRU	Annual capacity
	27-30	I4	IYR(1)	Year of first lump payment
	31-40	F10.0	LMPPAY(1)	Amount of first lump payment
	47-50	I4	IYR(2)	Year of second lump payment (if NLUMP = 2 or 3)
	51-60	F10.0	LMPPAY(2)	Amount of second payment
	67-70	I4	IYR(3)	Year of third lump payment (if NLUMP = 3)
	71-80	F10.0	LMPPAY(3)	Amount of third payment

Note: The number of cards type #1A or #1B must correspond to NCPENT in the preceding transfer station header card.

Card Type	Columns	Format	FORTRAN Name	Description
#2	1-3	I3	IDNUM	Numeric label for transfer station must match TRFIDN(J) in preceding header card
	11-20	F10.0	THRU	Annual throughput
	21-30	F10.0	TROPC	Annual operating cost

Note: The number of cards of type #2 must correspond to NOPENT in the preceding transfer station header card.

4.2.5 Plant Subfile

This subfile gives details of each existing or potential plant (incinerator or reclamation plant). The header card gives a numeric label, name, and upper and lower limit on throughput (if used) and the status of the plant for the particular run of the model. By setting the status to 0 (not used) a plant can be effectively excluded from a model run without any other changes to the data file.

Following each plant header card are one or more cards each giving the capital cost which would be incurred if the plant has a particular size. After this are one or more cards giving the annual operating cost which would be incurred if the plant has a particular throughput. The number of these cards is given by the header card. Finally a card gives the percentage of waste lost in processing and the percentage of waste recoverable as resource 1, 2, 3 etc. Data is read in for NUMPLT plants as given by card #4 of the general parameter subfile.

Card Type	Columns	Format	FORTRAN Name	Description
#0 (plant header card)				
	1-3	I3	PLTIDN(J)	Numeric label for plant
	4-11	A8	PLTNAM(J)	Name of plant
	15	I1	PLSTAT(J)	Status in this run 0 - not used 1 - used 2 - model decides
	21-30	F10.0	PLTLLT(J)	Lower limit on annual throughput (if used)
	31-40	F10.0	PLTULT(J)	Upper limit on annual throughput (if used)
	45	I1	IWAY	Way of entering capital costs 0 - amortization 1 - lump payment
	50	I1	NCPENT	number of different size plants for which capital data entered

Card Type	Columns	Format	FORTRAN Name	Description
	55	I1	NOPENT	Number of cost figures entered for different operating throughput
#1A (if IWAY = 0)				
	1-3	I3	IDNUM	Numeric label for plant -must match PLTIDN(J) in preceding header card
	11-20	F10.0	THRU	Annual capacity
	27-30	I4	ENDPAY	Last year of amortization payments
	31-40	F10.0	AMTPAY	Annual amortization payments
#1B (if IWAY = 1)				
	1-3	I3	IDNUM	Numeric label for plant -must match PLTIDN(J) in preceding header card
	10	I1	NLUMP	Number lump payments (1, 2 or 3)
	11-20	F10.0	THRU	Annual capacity
	27-30	I4	IYR(1)	Year of first lump payment
	31-40	F10.0	LMPPAY(1)	Amount of first lump payment
	47-50	I4	IYR(2)	Year of second lump payment (if NLUMP = 2 or 3)
	51-60	F10.0	LMPPAY(2)	Amount of second payment
	67-70	I4	IYR(3)	Year of third Lump lump payment (if NLUMP = 3)
	71-80	F10.0	LMPPAY(3)	Amount of third payment

Note: The number of cards type #1A or 1B must correspond to NCPENT in the preceding plant header card.

Card Type	Columns	Format	FORTRAN Name	Description
#2	1-3	I3	IDNUM	Numeric label for plant -must match PLTIDN(J) in preceding header card

Card Type	Columns	Format	FORTRAN Name	Description
	11-20	F10.0	THRU	Annual throughput
	21-30	F10.0	TROPC	Annual operating cost

Note: The number of cards of type #2 must correspond to NOPENT in the preceding plant header card.

#3	1-3	I3	IDNUM	Numeric label for plant - must match PLTIDN(J) in preceding header card
	11-15	F5.2	PLTLOS(J)	% of waste lost in processing
	16-20	F5.2	PLTRRN(J,1)	% of waste entering plant J recoverable as resource 1
	21-25	F5.2	PLTRRN(J,2)	2
	26-30	F5.2	PLTRRN(7,3)	3
	31-35	F5.2	PLTRRN(J,4)	4
	36-40	F5.2	PLTRRN(J,5)	5
	41-45	F5.2	PLTRRN(J,6)	6
	46-50	F5.2	PLTRRN(J,7)	7
	51-55	F5.2	PLTRRN(J,8)	8
	56-60	F5.2	PLTRRN(J,9)	9

Note: Any of the fields in card #3 may be blank, or zero.

4.2.6 Landfill Subfile

This subfile gives details of each existing or potential landfill. The header card gives a numeric label, name, an upper and lower limit on throughput (if used), the status of the landfill for this particular run and a limit on site capacity. By setting the status to 0 (not used), a landfill can be effectively excluded from a model run without any other changes to the data file.

Following each landfill header card are one or more cards each giving the capital cost which would be incurred if the landfill has a particular size. After this are one or more cards giving the annual operating cost which would be incurred if the landfill has a particular throughput. The number of these cards is given by the header card. Data is read in for NUMLFL landfills as given by card #4 of the general parameter subfile.

Card Type	Columns	Format	FORTRAN Name	Description
#0 (landfill header card)				
	1-3	I3	LFLIDN(J)	Numeric label for landfill
	4-11	A8	LFLNAM(J)	Name of landfill
	15	I1	LFSTAT(J)	Status in this run 0 - not used 1 - used 2 - model decides
	21-30	F10.0	LFLLLT(J)	Lower limit on annual throughput (if used)
	31-40	F10.0	LFLULT(J)	Upper limit on annual throughput (if used)
	45	I1	IWAY	Way of entering capital costs 0 - amortization 1 - lump payments
	50	I1	NCPENT	Number of different size landfills for which capital data entered
	55	I1	NOPENT	Number of cost figures entered for different operating throughput
	56-65	F10.0	LFMOST(J)	Greatest tonnage allowed over total period
#1A (if IWAY = 0)				
	1-3	I3	IDNUM	Numeric label for landfill- must match LFLIDN(J) in preceding header card
	11-20	F10.0	THRU	Annual capacity
	27-30	I4	ENDPAY	Last year of amortization payments
	31-40	F10.0	AMTPAY	Annual amortization payments
#1B (if IWAY - 1)				
	1-3	I3	IDNUM	Numeric label for landfill- must match LFLIDN(J) in preceding header card

Card Type	Columns	Format	FORTRAN Name	Description
	10	I1	NLUMP	Number lump payments (1, 2 or 3)
	11-20	F10.0	THRU	Annual capacity
	27-30	I4	IYR(1)	Year of first lump payment
	31-40	F10.0	LMPPAY(1)	Amount of first lump payment
	47-50	I4	IYR(2)	Year of second lump payment (if NLUMP= 2 or 3)
	51-60	F10.0	LMPPAY(2)	Amount of second payment
	67-70	I4	IYR(3)	Year of third lump payment (if NLUMP = 3)
	71-80	F10.0	LMPPAY(3)	Amount of third payment

Note: The number of cards type #1A or #1B must correspond to NCPENT in the preceding landfill header card.

Card Type	Columns	Format	FORTRAN Name	Description
#2	1-3	I3	IDNUM	Numeric label for landfill must match LFLIDN(J) in preceding header card
	11-20	F10.0	THRU	Annual throughput
	21-30	F10.0	TROPC	Annual operating cost

Note: The number of cards of type #2 must correspond to NOPENT in the preceding landfill header card.

4.2.7 Market Subfile

This subfile gives the name of each resource and details of each market location. The counts of these are given by NRESRC and NUMMKT(1), NUMMKT(2) etc. in card #4 and 5 of the general parameter subfile.

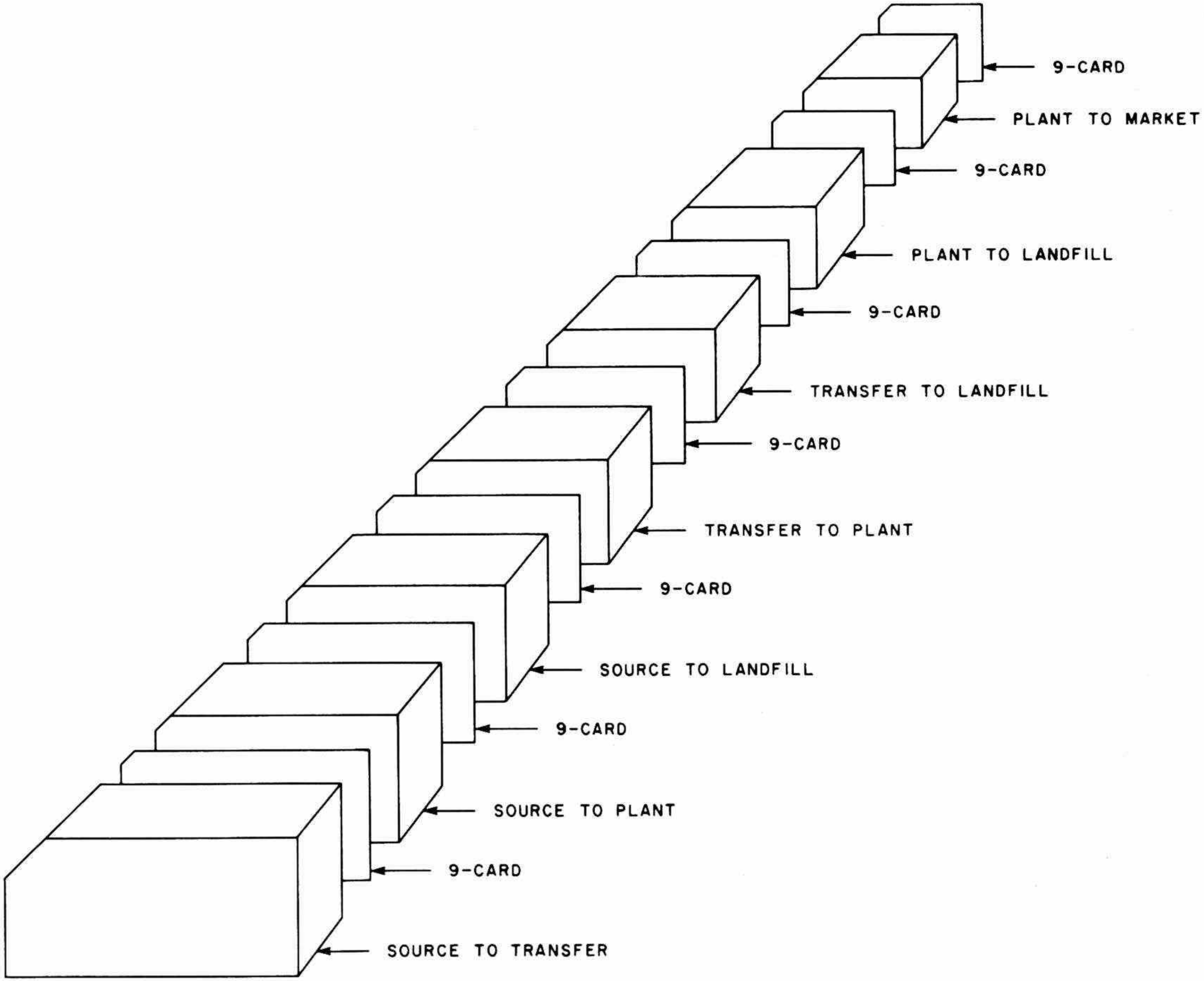
Card Type	Columns	Format	FORTRAN Name	Description
#0 (Resource header card)				
	1-8	A8	RSCNAM(J)	Name of resource J
#1 (One card for each location purchasing resource in preceding header card)				
	1	I1	JCODE	Resource number
	2-3	I2	KCODE	Numeric label for market location
	4-11	A8	MKTNAM(J,K)	Name for market location
	21-30	F10.3	PRICE	Price payable per tonne F.O.B. market location
	31-40	F10.0	DEMAND	Annual demand for resource at this location

4.2.8 Transportation Link Subfile (One card for each transportation link)

Every link which is available for transportation of material must be included in this subfile (Fig. 5). The number of links is not given in any previous card, but instead all links of one kind e.g. source to transfer station are grouped together and ended with a delimeter card (a card with '9' punched in column 1). If there are no links of a particular kind, the empty section ends with a delimeter.

Card Type	Columns	Format	FORTRAN Name	Description
#1	1	I1	ICODE	Code for type of link 1 - Source to transfer 2 - Source to plant 3 - Source to landfill 4 - Transfer to plant 5 - Transfer to landfill 6 - Plant to landfill 7 - Plant to market location

FIGURE 5 - ORGANISATION OF TRANSPORTATION LINK SUBFILE



Card Type	Columns	Format	FORTRAN Name	Description
				9 - change in type delimits above groups and end of file
	3-5	I3	IDORIG	Numeric label for origin
	8-10	I3	IDDEST	Numeric label for destination
	15	I1	IDVEIC	Number of vehicle used
	16-20	F5.0	TIME	Round trip time (minutes)

4.3 Error Messages From The Edit And Setup Program

The Edit and Setup Program performs some checking of the input data; for example, whether status of facilities have meaningful values and numeric labels are consistent between different parts of the input data. If such an error is detected, a message and error number is printed. The conditions causing each error number are listed below.

The most common mistake is an inconsistency between a count given in one card and the data to which the count refers. In many circumstances errors in the data will cause the Edit and Setup Program to stop (fatal FORTRAN error). It is usually easy to identify the card causing this by reading the printout from the Edit and Setup Program which lists information from each card as it is read in. The offending card is then the one following the last card from which information was printed.

The Edit and Setup Program makes very few checks for reasonableness of data (this varies so much from one study to another). The user should carefully check the printout of data read in, and intermediate calculated values, prior to running the entire model.

The following is a complete list of the error codes and their reasons in order of subroutine.

SUBROUTINE	ERROR CODE	REASON FOR ERROR
SOURCE	201	Variable IGENTP for type of generation data is less than 1 or greater than 5
	212	Variable IDNUMB for source number does not match SRCIDN(J) in preceding source header card
	223	First year entered in interpolation data does not match RUNBEG
	224	Last year entered in interpolation data does not match RUNEND
TRANFR	301	Variable TRSTAT(J) for status of transfer station is less than 0 or greater than 2
	302	Variable TRFLLT(J) for lower limit on annual throughput is greater than TRFULT(J) for upper limit
	311	Variable IDNUM does not match TRFIDN(J) in preceding transfer station header card
PLANTS	401	Variable PLSTAT(J) for status of plant is less than 0 or greater than 2
	402	Variable PLTLLT(J) for upper limit on annual throughput is greater than PLTULT(J) for upper limit
	411	Variable IDNUM does not match PLTIDN(J) on preceding plant header card
LANDFL	501	Variable LFSTAT(J) for status of landfill is less than 0 or greater than 2
	502	Variable LFLLLT(J) for lower limit on annual throughput is greater than LFLULT(J) for annual upper limit
	503	Variable LFLLLT(J) for lower limit on annual throughput is greater than LFMOST/RUNYRS (greatest amount over period/length of period)
	511	Variable IDNUM does not match LFLIDN(J) on preceding landfill header card
MARKET	601	First digit of MKTIDN does not match number for this resource

SUBROUTINE	ERROR CODE	REASON FOR ERROR
LINKEM	701	ICODE not 1 as required for source to transfer station links - card ignored
	702	ICODE not 2 as required for source to plant links - card ignored
	703	ICODE not 3 as required for source to landfill links - card ignored
	704	ICODE not 4 as required for transfer station to plant links - card ignored
	705	ICODE not 5 as required for transfer station to landfill links - card ignored
	706	ICODE not 6 as required for plant to landfill links - card ignored
	707	ICODE not 7 as required for plant to market links - card ignored
	711	IDORIG less than 100 - card ignored
	712	IDDEST less than 100 - card ignored
	721	Vehicle code less than 1 or greater than 8 - card ignored
	731	IDORIG does not match any SRCIDN in source data
	732	IDORIG or IDDEST does not match any TRFIDN in transfer station data
	733	IDORIG or IDDEST does not match any PLTIDN in plant data
	734	IDDEST does not match any LFLIDN in landfill data
	735	IDDEST does not match any MKTIDN in market data

4.4 Input For The Report Writer Program

The user can obtain reports for a single facility, for all facilities of a given type or for all facilities. This is specified on the first card.

The second card provides the means to obtain subtotals of costs for subgroups specified. In this way, if the study area consists of several municipalities, each municipality's portion of the total costs can be obtained. If subtotals are not required the card is left blank.

Card Type	Columns	Format	FORTRAN Name	Description
#1	1-8	A8	GENNAM	Name of generation area to be reported or ALL.
	9-16	A8	FACNAM	Name of facility to be reported or ALL
	17-20	A4	FACTYP	Type of facility to be reported or ALL
#2	1-5	I5	NOSUBT	Number of subtotals required
	6-10	I5	IDSUBT(1)	Numeric label of source to which first subtotal runs
	11-15	I5	IDSUBT(2)	Numeric label of source to which second subtotal runs
	16-20	I5	IDSUBT(3)	.
	.	.	.	.
	.	.	.	.
	.	.	.	.
	.	.	.	.
	.	.	.	.
	.	.	.	.
	71-75	I5	IDSUBT(14)	Numeric label of source to which fourteenth subtotal runs
	76-80	I5	IDSUBT(15)	Numeric label of source to which fifteenth subtotal runs.

CHAPTER FIVE

STEPS IN RUNNING THE MODEL

This chapter outlines the steps which should be taken in the early stages of a study to check that data are correct and the programs are set up properly. The data may be correct in the sense that all transcription and keypunch errors have been removed, but may still need amending if the user omitted potential facilities or desirable transportation links which will help in developing a low cost SWMS. This is likely to be recognized only after the model has been run a few times.

The following is suggested as a step-by-step procedure to achieve maximum assurance of correctness at low cost. The steps are:

1. Run the ESP alone until the data appear correct.
2. Make preliminary runs of the MIP to ascertain a solution is possible with the data used and the JCL and MIP control program are correct.
3. Run the whole model (ESP, MIP and RWP) to ensure it is operating correctly and the data allows reasonable solutions.

5.1 Checking the Data with the Edit and Setup Program

As a first step, run the ESP alone and check the printed output to see the data are correct. Card 3 of the general parameter subfile should include the following settings:

INREP = 2
OUTREP = 2
TWRITE = 0
DWRITE = 0
LANDCD = 0

The first two settings give maximum printed output. They can be set back to 1 or 0 in later runs when the user is satisfied the data are correct. TWRITE and DWRITE are set to 0 until the data are correct. In any case, if ESP detects any errors in the input data, the program sets these two switches to 0. LANDCD should be set to 0 for all runs.

ESP is run until all errors detected have been corrected. The user should then check the output very carefully. Inspect the printout of the raw data to check that the figures are as intended. The intermediate values calculated from the raw data should be reasonable.

5.2 First Run of the MIP

Once the user is satisfied the data are as intended, he can rerun ESP with TWRITE = 1 and DWRITE = 1. Check for the message 'WRITE TO T FILE COMPLETED' to ensure the file for input to MIP is created properly. If the message is not written by ESP and TWRITE = 1, then there is probably a JCL error.

For the first run of MIP, set TRSTAT, PLSTAT and LFSTAT to 1 (used) for every facility in the data base. With these settings the MIP should run at low cost and quickly find an optimal solution. If it fails to find an optimal solution when all the facilities are fixed as available, then the data needs revising. This run also provides a quick check that the JCL and MIP control program are correct.

5.3 First Run of the RWP

When the first run of the MIP has been successful, the MIP solution file will be created on disk. The RWP reads this file, DFILE and two input cards. If these files are all made available to the RWP then it should run successfully. Any problems are likely to be caused by incorrect JCL or empty files.

5.4 'Production' Runs of the Model

Once the user is satisfied the model is set up correctly and the data file is free of transcription and keypunch errors, he can start the analysis runs. For these the setting of flags in card 3 of the general parameter subfile should be INREP = 1, OUTREP = 1, TWRITE = 1 and DWRITE = 1. Although each of the three programs ESP, MIP and RWP can be run at different times by saving intermediate files, care should be taken for a particular set of data values that the printed output from each of the three programs be kept together.

The sequence in which different scenarios are run is unlikely to be known completely in advance, since it may depend on results found in early runs. A valuable guideline in early runs is for most of the facilities to be pegged as 'used' or 'not used'. This will keep down computer costs while the user is still confirming data and becoming familiar with the form and cost of feasible solutions.

PART III

PROGRAMMER'S GUIDE

CHAPTER SIX

RUNNING THE MODEL ON A CDC COMPUTER

This chapter gives information on how to set up and run the version of the model written for a CDC computer using the package APEX-III.

6.1 Setting Device Numbers, Date and Dimension Statements in the ESP

The ESP uses one input file and three output files (Fig. 6). The files, allocations and current device numbers are:

ESP input data on unit R. Currently set R=5.

ESP printout on unit W. Currently set W=6.

TFILE to enter APEX-III on unit T. Currently set T=1.

DFILE passed to RWP on unit D. Currently set D=9.

The PROGRAM statement is consistent with the above settings. If the user wishes to change file allocations, this may be done by altering the assignment of values to R,W,T,D in the main program, and making appropriate changes in the PROGRAM statement.

The internal date is called in the main routine and written out in subroutine HEADER with format A10. The call and format may be different in some installations.

Some of the implied dimensioning in labelled COMMON may need altering for some data bases:

Each array in SRCDAT needs one element per source.

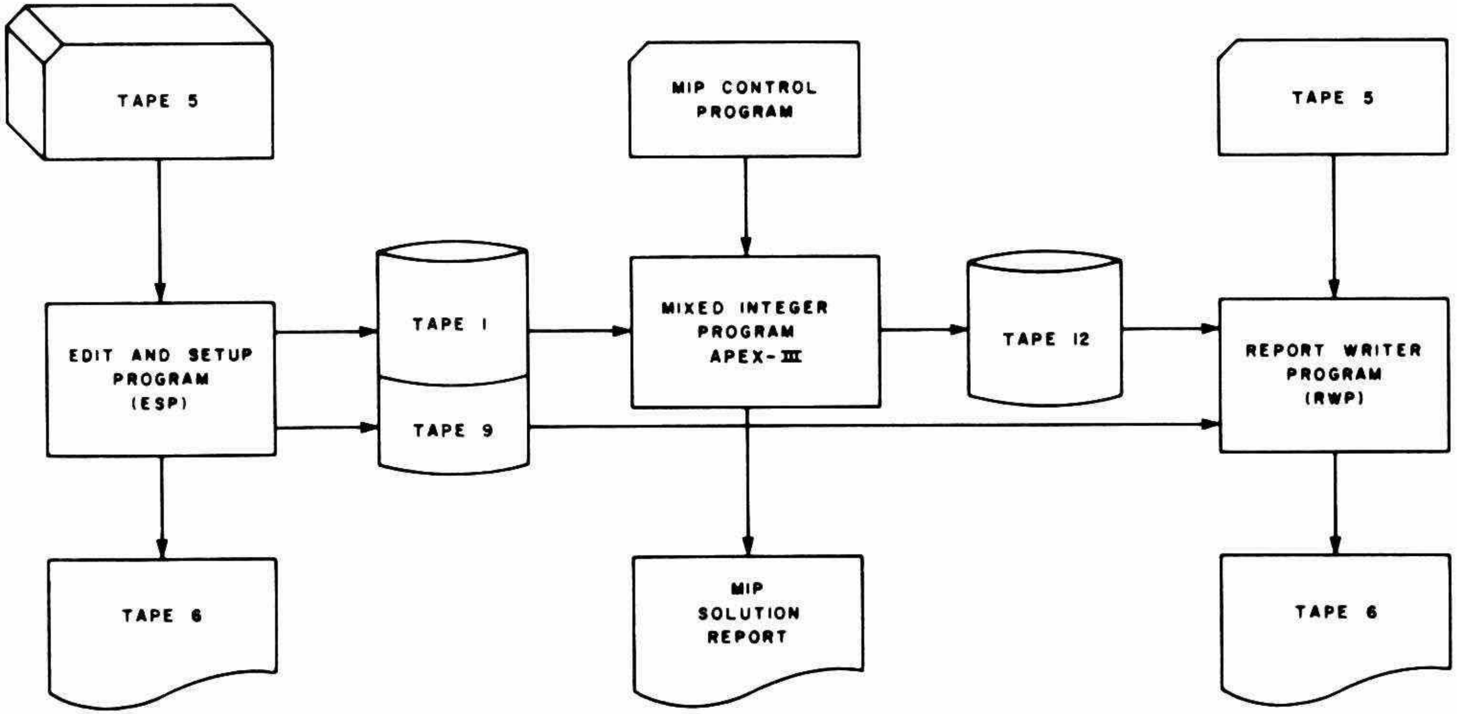
Each array in TRFDAT needs one element per transfer station.

Each array in PLTDAT needs one element per plant.

Each array in LFLDAT needs one element per landfill.

Each array in LNKDAT needs one element per transportation link.

FIGURE 6 - FILE NAMES USED IN APEX-III VERSION


TAPE 5
MIP CONTROL PROGRAM
TAPE 5
EDIT AND SETUP PROGRAM (ESP)
TAPE 1
TAPE 9
MIXED INTEGER PROGRAM APEX-III
TAPE 12
REPORT WRITER PROGRAM (RWP)
TAPE 6
MIP SOLUTION REPORT
TAPE 6

6.2 Setting Device Numbers, Lines Per Page, Date and Dimension Statements in the RWP

The RWP uses three input files and one output file (Fig. 6). The files,allocations, and current device numbers are:

RWP input cards currently set on TAPE 5

MIP solution file currently set on TAPE 12

DFILE passed from RWP currently set on TAPE 8

RWP printout currently set on TAPE 6

The PROGRAM statement is consistent with these settings. If the user wishes to change file allocations, this may be done by altering the READ statements in the main program and making appropriate changes in the PROGRAM statement.

The variable LINPAG should be set in the main program to the number of lines per page for the RWP Printout.

The internal date is called in the main program and written out in subroutine HEADER with format A10. The call and format may be different in some installations.

Some of the implied dimensioning in labelled COMMON may need altering for some data bases:

Each array in SRCDAT needs one element per source.

Each array in FACDAT should be dimensioned identically and allow one element per facility of each type

Each array in LNKDAT needs one element per transportation link.

In COMMON area OUT, array ARCWST should have one element per transportation link and the arrays WST and DELTA should have the first dimension the same as arrays in FACDAT.

While not essential, it is simplest and safest to make dimensioning identical between similar arrays in the ESP and the RWP.

6.3 JCL for Running on CDC with APEX-III

The JCL given here is for running each of the three programs as separate jobs. All files are read and written sequentially. Obviously, if the programs are run consecutively as part of the same job, some of the CATALOG and ATTACH statements should be omitted. Figure 7 gives the JCL for running the three programs under SCOPE (2) while Figure 8 gives the JCL for runnning under NOS (3).

6.4 The Control Program for APEX-III

Figure 9 lists the APEX-III control program to use when running the model. It is identical whichever operating system is used. In the listing RPOPTCR is set to a value of 0.01. This means the Branch-and-Bound search will step when a solution is found proven to be within 1% of the global optimum. Experience shows that the value of 0.03 is a reasonable setting for finding a good or best solution at reasonable computer cost. See the APEX-III manual (1) for further information.

FIGURE 8 - JCL FOR RUNNING ON CDC UNDER NOS

```
00100 GET, TAPE5=P78.
00110 DEFINE, TAPE9=T9R1.
00120 DEFINE, TAPE1=T1R1.
00130 SBULIM, JS=250.
00140 GET, JPSET.
00150 FTN, I=JPSET, L=0, GO.
00160 REWIND, TAPE6.
00170 DAYFILE, JPDAY.
```

EDIT AND SETUP PROGRAM

```
00100 /JOB
00110 REPORT, T100, P4.
00120 /USER
00130 DEFINE, OUTPUT=JPOUTR.
00140 GET, JPRB.
00150 ATTACH, TAPE8=T9R1A.
00160 GET, TAPE5=RPJAN15.
00170 ATTACH, TAPE12=T12R1.
00190 JPRB.
00200 DAYFILE.
00210 EXIT.
00220 DAYFILE.
00230 /EOF
```

REPORT WRITER PROGRAM

```
00100 /JOB
00110 RUNAPEX, T300, P4.
00120 /USER
00140 DEFINE, OUTPUT=OUTAPEX.
00150 ATTACH, APEX/UN=LIBRARY.
00160 ATTACH, TAPE1=T1R1A.
00170 DEFINE, TAPE12=T12R1.
00180 RFL, 200000.
00185 REDUCE(-).
00190 APEX (S, MIN, OBJ=NETCOST, MIP, FS)
00200 DAYFILE.
00210 EXIT.
00220 DAYFILE.
00230 /EOF
```

CONTROL PROGRAM FOR APEX-III

FIGURE 7 - JCL FOR RUNNING ON CDC UNDER SCOPE

```
00100 /JOB
00110 TEST, T100, P4.
00120 USER, MOESSCC, PASWORD.
00130 REQUEST, TAPE1, *PF.
00140 REQUEST, TAPE9, *PF.
00150 FTN, L=0.
00155 MAP, OFF.
00160 LGO, TAPE1, INPUT, OUTPUT.
00170 REWIND, TAPE1.
00180 CATALOG, TAPE1, APEXINPUT1, NA=1.
00190 CATALOG, TAPE9, REPINPUT1, NA=1.
00200 /EOR
00210 /NOSEQ
00220 /READ, OUTREP
00230 /EOR
00240 /READ, S3C1P4C
00250 /EOR
00260 /EOF
```

EDIT AND SETUP PROGRAM

```
00100 /JOB
00110 REPORT, T100, P4.
00120 USER, MOESSCC, PASWORD.
00142 FTN, L=0.
00144 MAP, OFF.
00160 ATTACH, TAPE8, REPINPUT1.
00170 ATTACH, TAPE12, APEXOUTPUT1.
00180 LGO, TAPE12, INPUT, TAPE8.
00190 /EOR
00200 /NOSEQ
00210 /READ, JPREP1
00215 /SEQ
00220 /EOR
00230 ALL    ALL
00240     9  112  126  153  159  164  177  253  261  267
00250 /EOR
```

REPORT WRITER PROGRAM

FIGURE 9 - CONTROL PROGRAM FOR APEX-III

```
00100 /JOB
00110 APEX,T100,P4.
00120 USER,MOESSCC,PASWORD.
00130 HEADING.B R.KIRBY
00140 HEADING.M MACLAREN
00150 REQUEST,TAPE12,*PF.
00160 APPLIC,APEX.
00170 ATTACH,TAPE1,APEXINPUT1.
00180 RFL,100000.
00190 APEX (S,MIN,N=ONTWAST,OBJ=NETCOST,PS,MIP)
00200 CATALOG,TAPE12,APEXOUTPUT,NA=1.
```

DEFAULT OPTIMIZATION VALUE OF 10%

```
00100 /JOB
00110 APEX,T1000,P4.
00120 USER,MOESSCC,PASWORD.
00125 SBULIM (2500)
00130 HEADING.B R.KIRBY
00140 HEADING.M MACLAREN
00150 REQUEST,TAPE12,*PF.
00160 APPLIC,APEX.
00170 ATTACH,TAPE1,APEXINPUT1.
00180 RFL,120000.
00190 APEX.
00200 CATALOG,TAPE12,APEXOUTPUT1,NA=1.
00210 /EOR
00220          INPUT      ONTWAST
00230          SET        KNDIR      MIN
00240          SET        KNOBJ      NETCOST
00250          SET        KNRHS      0
00252          SET        KNBND      0
00254          SET        KNRNG      0
00260          SET        RPOPTCR    0.01
00270          SELECT
00272          REDUCE
00280          CRASH
00290          PRIMAL
00300          MIXINT
00302          RECREATE
00304          PRIMAL
00310          OUTPUT     PART/SPEC
00320          EXIT
00330 /EOR
```

OPTIMIZATION VALUE OF 1%

CHAPTER SEVEN

RUNNING THE MODEL ON AN IBM COMPUTER

This chapter gives information on how to set up and run the version of the model written for an IBM/370 computer using the package MPSX-MIP.

7.1 Setting Device Numbers, Date and Dimension Statements in the ESP

The ESP uses one input file and three output files (Fig. 10). The files, allocations and current device numbers are:

ESP input Data on unit R. Currently set R=5.

ESP printout on unit W. Currently set W=6.

TFILE to enter MPSX-MIP on unit T. Currently set T=1.

DFILE passed to RWP on unit D. Currently set D=9.

If the user wishes to change file allocations, this may be done by altering the assignment of values to R,W,T,D in the main program and amending the FT..FOO1 file names in the JCL DD statements.

The internal date is called in the main routine and written out in subroutine HEADER. The call and format should be altered to match those used at the installation where the model is run.

The implied dimensioning in labelled COMMON may need altering for some data bases:

Each array in SRCDAT needs one element per source.

Each array in TRFDAT needs one element per transfer station.

Each array in PLTDAT needs one element per plant.

Each array in LFLDAT needs one element per landfill.

Each array in LNKDAT needs one element per transportation link.

7.2 Setting Device Numbers, Lines per Page, Date and Dimension Statements in the RWP.

The RWP uses three input files and one output file (Fig. 10). The files, allocation, and current device numbers are:

RWP input card on unit R. Currently set R=5.

MIP solution file on unit F. Currently set F=12.

DFILE passed from RWP on unit D. Currently set D=9.

RWP Printout on unit W. Currently set W=6.

If the user wishes to change file allocations, this may be done by altering the assignment of values to R,W,T,D in the main program and amending the FT..FOO1 file names in the JCL DD statements.

The variable LINPAG should be set in the main program to the number of lines per page for the RWP Printout.

The internal date is called in the main program and written out in subroutine HEADER. The call and format should be altered to match those used at the installation where the model is run.

The implied dimensioning in labelled COMMON may need altering for some data bases:

Each array in SRCDAT needs one element per source

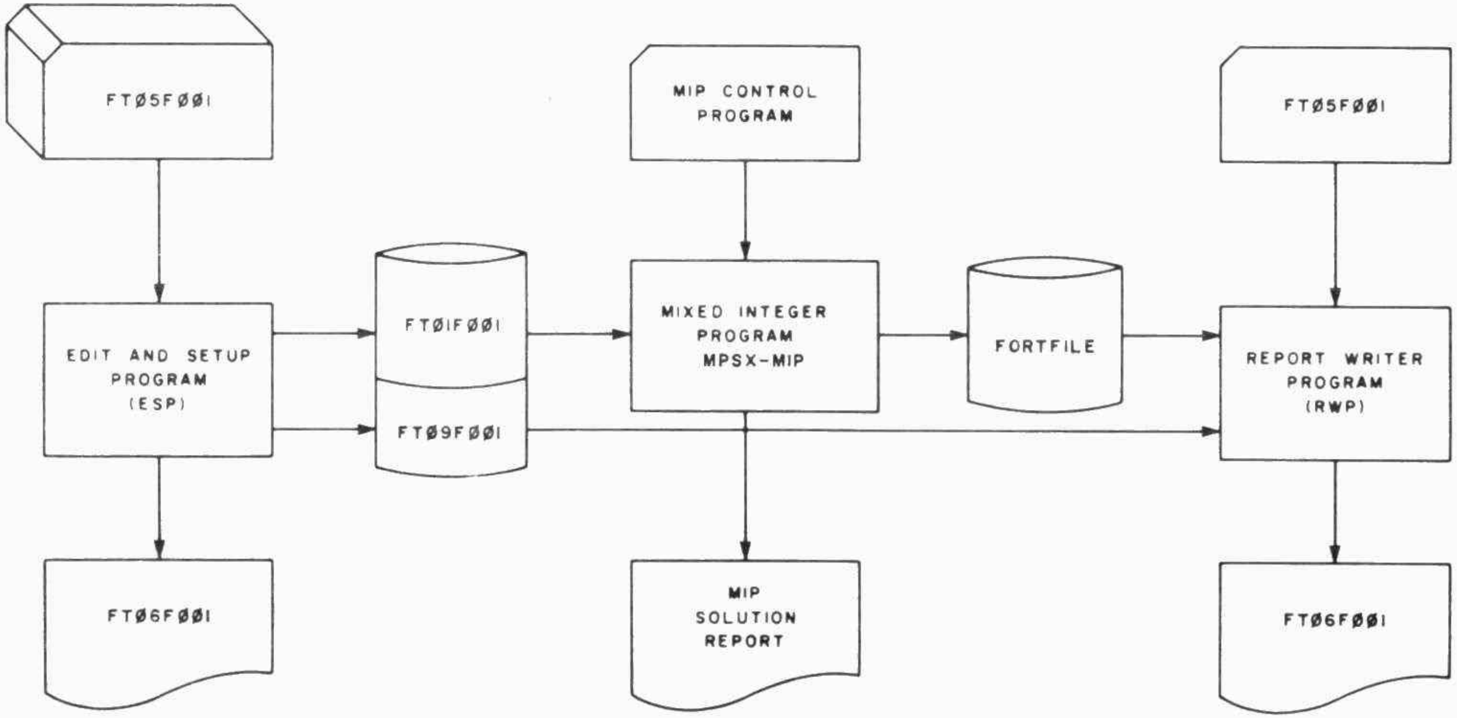
Each array in FACDAT should be dimensioned identically and allow one element per facility of each type.

Each array in LNKDAT needs one element per transportation link.

In COMMON area OUT, array ARCWST should have one element per transportation link and the arrays WST and DELTA should have the first dimension the same as arrays in FACDAT.

While not always essential, it is simplest and safest to make dimensioning idential between similar arrays in the ESP and the RWP.

FIGURE 10 - FILE NAMES USED IN MPSX-MIP VERSION



7.3 JCL for Running on IBM with MPSX-MIP

The JCL given here is for running each of the three programs as separate jobs. All files are read and written sequentially. Obviously if the programs are run consecutively as part of the same job, files may be passed rather than kept. Figure 11 gives the JCL for running the three programs under OS.

7.4 The Control Program for MPSX-MIP

Figure 12 lists the MPSX-MIP control program to use when running the model. The macro call OPTIMIX can include in its argument list a limit on CPU time (the third parameter, unit is minutes, 0 indicates no time limit). The command SOLUTION 'FORTFILE' writes the best solution to the data set FORTFILE from which it can be read by RWP.

FIGURE II - JCL FOR RUNNING ON IBM/370 UNDER OS

```
//JOB1 JOB
//EXEC FORTHCLG
//FORT.SYSIN DD *

                   Edit and Setup Program

/*
//GO.FT05F001 DD *

                   Model Data

/*
//GO.FT06F001      DD SYSOUT=A
//GO.FT08F001      DD DSN=TFILE,DISP=(NEW,KEEP),
//                 DCB=(RECFM=FB,LRECL=80,BLKSIZE=3520),
//                 SPACE=(TRK,(10,10),RLSE),UNIT=3330,VOL=SER=WORK30
//GO.FT09F001      DD DSN=DFILE,DISP=(NEW,KEEP),
//                 DCB=(RECFM=VBS,RECL=204,BLKSIZE=1024),
//                 SPACE=(TRK,(1,1),RLSE),UNIT=3330,VOL=SER=WORK30
//


//JOB2 JOB
//EXEC MPSXMIP
//MIP.SYSIN DD *

                   Control Program

/*
//GO.SYSIN DD      DSN=TFILE,DISP=OLD,
//                 DCB=(RECFM=FB,LRECL=80,BLKSIZE=3520),
//                 UNIT=3330,VOL=SER=WORK30
//GO.FORTFILE      DD DSN=FORTFILE,DISP=(NEW,KEEP),
//                 DCB=(RECFM=VBS,LRECL=204,BLKSIZE=1024),
//                 SPACE=(CYL,(1,1)),UNIT=3330,VOL=SER=WORK30
//

//JOB3 JOB
//EXEC FORTHCLG
//FORT.SYSIN DD *

                   Report Writer Program

/*
//GO.FT05F001 DD *

                   Subtotal Request Card

//GO.FT06F001      DD SYSOUT=A,DCB=(RECFM=FA,BLKSIZE=133,LRECL=133)
//GO.FT09F001      DD DSN=DFILE,DISP=OLD
//                 DCB=(RECFM=VBS,LRECL=204,BLKSIZE=1024),
//                 UNIT=3330,VOL=SER=WORK30
//GO.FT12F001      DD DSN=FORTFILE,DISP=OLD,
//                 DCB=(RECFM=VBS,LRECL=204,BLKSIZE=1024),
//                 UNIT=3330,VOL=SER=WORK30
//
```

FIGURE 12 - CONTROL PROGRAM FOR MPSX-MIP

```
//MIP.SYSIN DD *
          PROGRAM
          INITIALZ
          MOVE(XDATA,'ONTWAST')
          MOVE(XPBNAME,'PROBFL')
          CONVERT
          SETUP('BOUND','FACILITY')
          MOVE(XOBJ,'NETCOST')
          MOVE(XRHS,'ORIGRHS')
          OPTIMIZE
          SOLUTION
          OPTIMIX('COST',0.,0,0,0)
          SOLUTION('FILE','FORTFILE')
          EXIT
          PEND
/*
```

CHAPTER EIGHT

DOCUMENTATION OF THE EDIT AND SETUP PROGRAM

This chapter documents, routine by routine, the ESP. The documentation applies equally to the versions for CDC and IBM.

The differences between the two versions are:

a) the PROGRAM statement is required only by the CDC version.

b) the DATE call may be different.

c) the variables typed DOUBLE PRECISION in the IBM version are typed REAL in the CDC version.

d) there are minor differences in a few of the WRITE statements preparing TFILE, according to whether it is written for APEX-III or MPSX-MIP.

The differences of type d) are explained in the descriptions of subroutine TSETUP. Figure 13 lists all the routines in their order of execution. Figure 14 shows which COMMON blocks are accessed by which routines.

8.1 Main Routine

In the main routine device numbers are set and the various subroutines are called.

8.1.1 Glossary of Important Variables

The following variables are all read in by routine START.

NUMTRF	-	number of possible transfer stations
NUMPLT	-	number of possible plants
NRESRC	-	number of different materials recoverable
TWRITE	-	flag to write TFILE
DWRITE	-	flag to write DFILE

FIGURE 13 - SEQUENCE FOR EXECUTION OF SUBROUTINES IN ESP

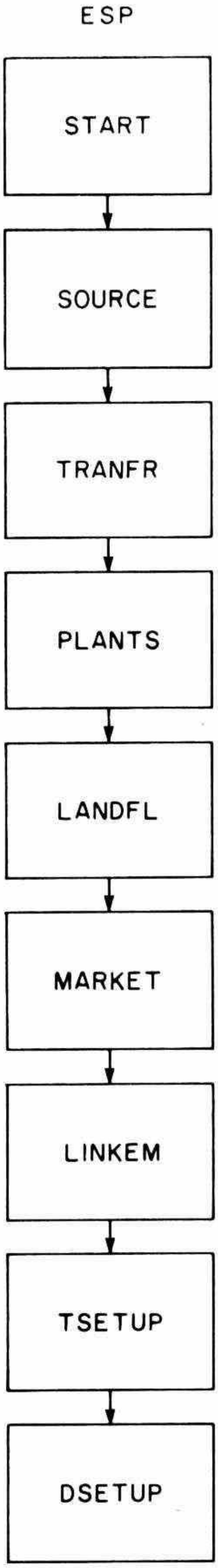


FIGURE 14 - COMMON BLOCKS NEEDED IN ESP

	BLOCK NAME							
ROUTINE	GENDAT	VECDAT	SRCDAT	TRFDAT	PLTDAT	LFLDAT	MKTDAT	ARCDAT
MAIN	☐							
HEADER	☐							
ERROR	☐							
START	☐	☐						
SOURCE	☐		☐					
TRANFR	☐			☐				
USPWF								
REGRES								
PLANTS	☐				☐			
LANDFL	☐					☐		
MARKET	☐						☐	
LINKEM	☐	☐	☐	☐	☐	☐	☐	☐
TSETUP	☐	☐	☐	☐	☐	☐	☐	☐
DSETUP	☐	☐	☐	☐	☐	☐	☐	☐

8.2 Subroutine START

This routine initializes some variables then reads in the general parameter subfile. If a printout is requested, all data are listed with appropriate explanation.

8.2.1 Glossary of Important Variables

The meaning of most of the variables is given on page 16 in the description of the general parameter subfile.

TOTERR - the total number of errors detected in the data by the ESP.

ERFLAG - if this flag has value 1, an error has been detected by ESP. Special action may be taken before the flag is reset to 0

RUNYRS - number of years in the study period.

8.2.2 Processing

Variables are read from the general parameter subfile into COMMON blocks GENDAT and VECDAT. If INREP is 1 or 2, the variables are listed in the Printout.

8.3 Subroutine SOURCE

This routine reads in ID number, name and details of waste generation for each source. Total amount of waste generated at each source is calculated for entry to the MIP, and may be printed. The routine is made up of a major DO loop iterated once for each source.

8.3.1 Glossary of Important Variables

NUMSRC - number of sources in the model (read in by routine START)

SRCIDN - ID number for source (3 digits)

SRCNAM - name of source (8 characters)

IGENTP - number 1 to 5 indicating form in which generation data is entered for this source

IDNUMB - ID number on card with generation data - should match immediately preceding SRCIDN

SRCTOT - total amount of waste generated at this source over the period

ALWAST - total amount of waste generated at all sources over the period

RUNYRS - number of years in the period

8.3.2 Calculations

If IGENTP = 2

The figures entered are AMONT1 (total generation in the first year) and GROWTH (% growth may be zero or negative)

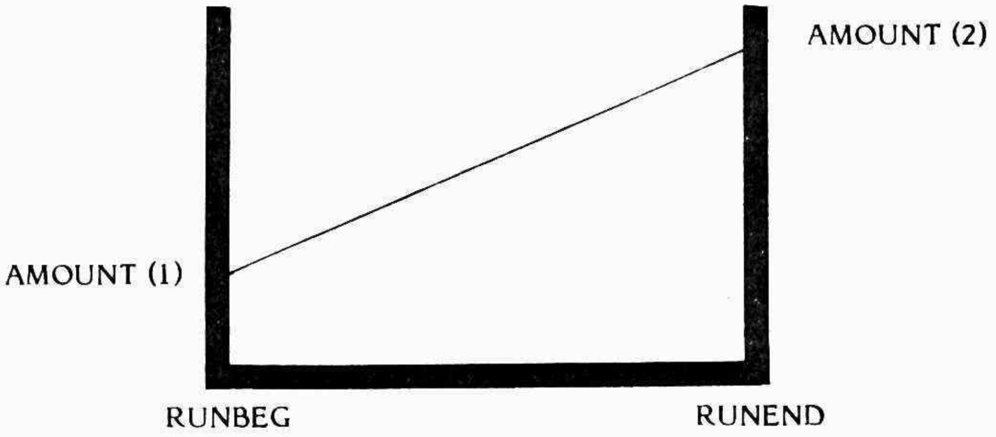
$$\text{SRCTOT} = \text{AMONT 1} + \text{AMONT1} \sum_{J=1}^{\text{NYRS}} (1 + \text{GROWTH}/100)^{J}$$

NYRS is 1 less than the length of the period

If IGENTP = 3

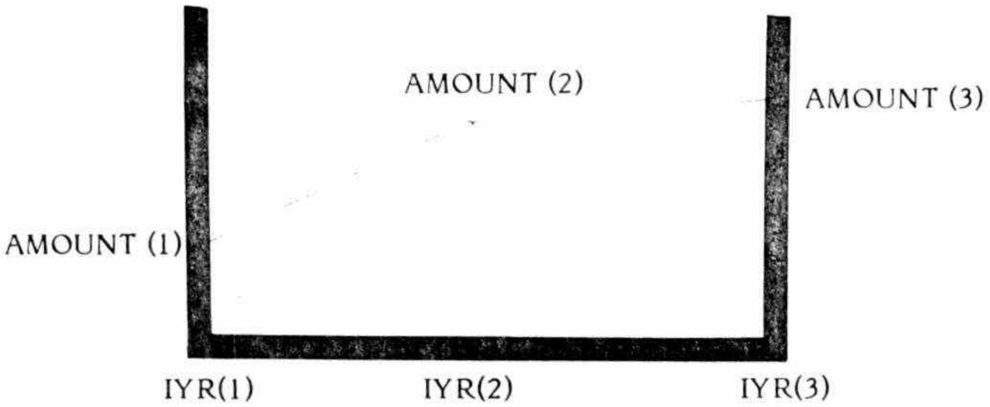
A total generation figure is entered for 2, 3, or 4 specified years of the period. The period total is calculated by interpolation. The first year entered must coincide with the first year of the period, and the last year entered must coincide with the last year of the period.

If data are entered for 2 years:



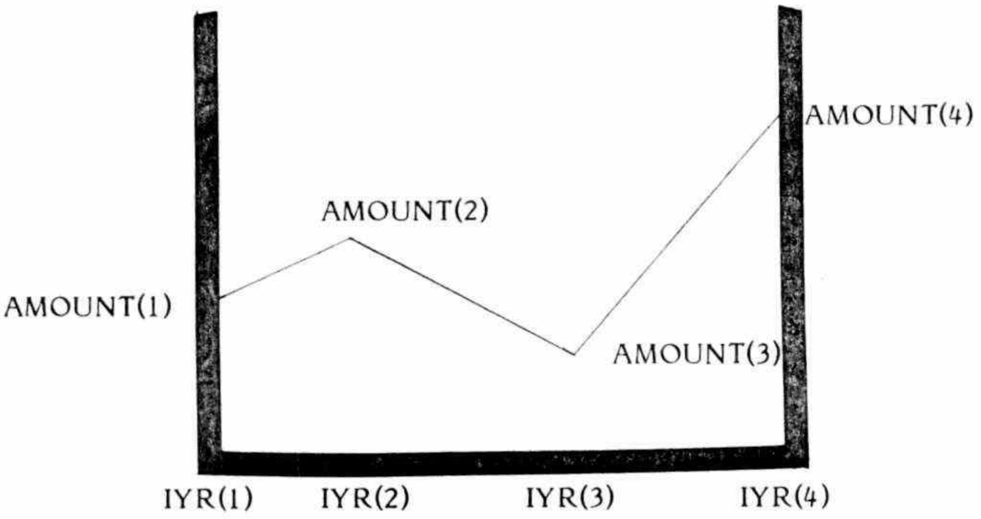
SRCTOT = (AMOUNT (1) + AMOUNT(2))/2 * RUNYRS

If data are entered for 3 years:



SRCTOT = (AMOUNT(1) + AMOUNT(2))/2 * (IYR(2) -IYR(1) + 1)
+(AMOUNT(3) + AMOUNT(2))/2 * (IYR(3) -IYR(2) + 1)
- AMOUNT(2)

If data are entered for 4 years:



SRCTOT = (AMOUNT(1) + AMOUNT(2))/2 * (IYR(2) -IYR(1) + 1)
+ (AMOUNT(2) + AMOUNT(3))/2 * (IYR(3) -IYR(2) +1)
+ (AMOUNT(3) + AMOUNT(4))/2 * (IYR(4) - IYR (3) +1)
- AMOUNT (2) -AMOUNT (3)

If IGENTP = 4

The figures entered are TONYR1 (tonnes per capita generation in the first year), TONGRO (% growth in generation per capita - this may be zero or negative), POPYR1 (population in the first year), POPGRO (% growth in population - may be zero or negative).

In the first year

$$\text{YRTOT} = \text{TONYR1} * \text{POPYR1}$$

$$\text{GROW} = (1 + \text{TONGRO}/100) * (1 + \text{POPGRO}/100)$$

Then

$$\text{SRCTOT} = \text{YRTOT} + \text{YRTOT} \sum_{J=1}^{\text{NYRS}} (\text{GROW})^{J}$$

NYRS is 1 less than the length of the period

If IGENTP = 5

This parallels IGENTP 3 but instead of entering total generation in a year the figures entered are TONYR(K) (tonnes per capita generation in year K) and POPYR (K) (population in year K). Amount in year K is then:

AMOUNT(K) = TONYR(K) * POPYR(K)

8.4 Subroutine TRANFR

This routine reads in ID number, name, capacity limits and capital and operating cost data for each transfer station. The cost figures to enter the MIP are developed by linear regression on the raw data entered. The routine is made up of a major DO loop iterated once for each transfer station

8.4.1 Glossary of Important Variables

NUMTRF	-	number of possible transfer stations in the model (read in by routine START)
TRFIDN	-	ID number for transfer station (3 digits)
TRFNAM	-	name of transfer station (8 characters)
TRSTAT	-	status of transfer station in this run 0 - not used 1 - used 2 - model decides

TRFLLT - lower limit on annual throughput (if used)
TRFULT - upper limit on annual throughput (if used)
IWAY - way of entering capital cost
    0 - amortization
    1 - lump payments
NCPENT - number of different size stations for which capital data entered
NOPENT - number of cost figures entered for different operating throughput

8.4.2 Calculations

Capital cost figures are entered either as an amortization stream, or as lump payments for the facility (according to IWAY). Each set of capital cost figures entered corresponds to a particular capacity of that particular facility. One card holds the data for each capacity.

To enter capital costs as an amortization stream for a particular capacity (THRU), the last year of payment (ENDPAY) and the annual payment (AMTPAY) are entered. The payment stream is discounted to an effective lump payment at the beginning of the study period by the formula:
COST = AMTPAY * USPWF (RATE, NYRS)
The function USPWF calculates a present worth factor and is documented on page 57.
RATE is the discount rate
NYRS = ENDPAY - RUNBEG + 1

To provide capital costs as lump payments, the user enters one, two or three payment amounts for a particular capacity of facility. The payments are discounted to an effective lump payment at the beginning of the study period.

Fixed and variable capital costs are calculated by calling the routine REGRES documented on page 66.

Operating costs are entered for one or more operating throughputs of the facility. One card holds the data for each throughput. Fixed and variable operating costs are calculated by the routine REGRES.

Since operating costs will be incurred in each year of the study period, they are multiplied by the factor USPWF (RATE, RUNYRS) before entering the MIP model.

8.5 Subroutine PLANTS

This routine reads in ID number, name, capacity limits, capital and operating cost and processing data for each plant. The cost figures to enter the MIP are developed by linear regression on the raw data entered. The routine is made up of a major DO loop iterated once for each plant.

8.5.1 Glossary of Important Variables

NUMPLT - number of possible plants in the model (read in by routine START)
PLTIDN - ID number for plant (3 digits)
PLTNAM - name of plant (8 characters)
PLSTAT - status of plant in this run
  0 - not used
  1 - used
  2 - model decides
PLTLLT - lower limit on annual throughput (if used)
PLTULT - upper limit on annual throughput (if used)
IWAY - way of entering capital costs
  0 - amortization
  1 - lump payments
NCPENT - number of different size plants for which capital data entered
NOPENT - number of cost figures entered for different operating throughput
PLTLOS - % of waste lost in processing
PLTKEP - fraction of waste kept in processing
PLTRRN(J,M)- % of waste recoverable as material M at this plant

8.5.2 Calculations

Capital cost figures are entered either as an amortization stream, or as lump payments for the facility (according to IWAY). Each set of capital cost figures entered corresponds to a particular capacity of that particular facility. One card holds the data for each capacity.

To provide capital costs as an amortization stream for a particular capacity (THRU), the last year of payment (ENDPAY) and the annual payment (AMTPAY) are entered. The payment stream is discounted to an effective lump payment at the beginning of the study period by the formula:

COST = AMTPAY * USPWF (RATE, NYRS)
RATE is the discount rate
NYRS = ENDPAY - RUNBEG + 1

To provide capital costs as lump payments the user enters one, two, or three payment amounts for a particular capacity of facility. The payments are discounted to an effective lump payment at the beginning of the study period.

Fixed and variable capital costs are calculated by calling the routine REGRES.

Operating costs are entered for one or more operating throughputs of the facility. One card holds the data for each throughput. Fixed and variable operating costs are calculated by the routine REGRES

Since operating costs will be incurred in each year of the study period, they are multiplied by the factor USPWF (RATE, RUNYRS) before entering the MIP model.

The variable PLTLOS is divided by 100 to convert it from a percentage to a fraction. Since the formulation of the model actually requires the proportion kept, PLTKEP is calculated as follows:

PLTKEP = - (1- PLTLOS/100)

The variables PLTRRN(J,M) for M = 1 to 9 are divided by 100 to convert them from percentages to fractions, then negated since in the formulation they are preceded by a minus sign.

8.6 Subroutine LANDFL

This routine reads in ID number, name, capacity limits and capital and operating cost data for each landfill. The cost figures to enter the MIP are developed by linear regression on the raw data entered. The routine is made up of a major DO loop iterated once for each landfill.

8.6.1 Glossary of Important Variables

NUMLFL	-	number of possible landfills in the model (read in by routine START)
LFLIDN	-	ID number for landfill (3 digits)
LFLNAM	-	name of landfill (8 characters)
LFSTAT	-	status of landfill in this run 0 - not used 1 - used 2 - model decides
LFLLLT	-	lower limit on annual throughput (if used)
LFLULT	-	upper limit on annual throughput (if used)
IWAY	-	way of entering capital costs 0 - amortization 1 - lump payments
NCPENT	-	number of different size landfills for which capital data entered
NOPENT	-	number of cost figures entered for different operating throughput
LFMOST	-	greatest total tonnage the landfill can receive over the total period.

8.6.2 Calculations

Capital cost figures are entered either as an amortization stream, or as lump payments for the facility (according to IWAY). Each set of capital cost figures entered corresponds to a particular capacity of that particular facility. One card holds the data for each capacity.

To provide capital costs as an amortization stream for a particular capacity (THRU), the last year of payment (ENDPAY) and the annual payment (AMTPAY) are entered. The payment stream is discounted to an effective lump payment at the beginning of the study period by the formula:

COST = AMTPAY * USPWF (RATE,MYRS)
RATE is the discount rate
NYRS = ENDPAY - RUNBEG + 1

To provide capital costs as lump payments, the user enters one, two, or three payment amounts for a particular capacity of facility. The payments are discounted to an effective lump payment at the beginning of the study period.

Fixed and variable capital costs are calculated by calling the routine REGRES documented later.

Operating costs are entered for one or more operating throughput of the facility. One card holds the data for each throughput. Fixed and variable operating costs are calculated by the routine REGRES.

Since operating costs will be incurred in each year of the study period, they are multiplied by the factor USPWF (RATE,RUNYRS) before entering the MIP.

The greatest amount of waste which can be shipped to landfill is taken as the lesser of the two values (LFLULT * RUNYRS) and LFMOST. This is the value stored in COMMON/LFLDAT/ for writing to TFILE.

8.7 Subroutine MARKET

This routine reads in market data, in order of resource.

8.7.1 Glossary of Important Variables

NRESRC - number of resources considered in the model (read in by routine START)
NUMMKT(J) - number of market locations for resource J (read in by routine START)
RSCNAM(J) - name of resource J (8 characters)

MKTIDN(J,K) - ID number for market location K at which resource J is sold (3 digits with first equal to J)

MKTNAM(J,K) - name for market location K at which resource J is sold (8 characters)

DEMAND - annual demand at market location K for resource J

PRICE - price per tonne payable at market location K for resource J

DEMAND(J,K) - total demand over period

PRICET(J,K) - effective price (after discounting) to be used in model

8.8 Subroutine LINKEM

This routine reads in the transportation subfile, checks that the origin and destination code for each arc (link) matches the ID code of a locations and stores details of the links in COMMON/LNKDAT/. According to the setting of the flag INREP, details of each link might be listed.

8.8.1 Glossary of Important Variables

ICODE - type of link

IDORIG - ID number of link origin

IDDEST - ID number of link destination

IDVEIC - type of vehicle which will transport along link

TIME - round trip travel time (minutes)

COST - cost to transport 1 tonne of material along link

ARCOST - discounted cost to transport 1 tonne of material

NANDST - total number of source to transfer links

NANDSP - total number of links up to and including all source to plant links

NANDSL - total number of links up to and including all source to landfill links

NANDTP - total number of links up to and including all transfer to plant links

NANDTL - total number of links up to and including all transfer to landfill links

NANDPL	-	total number of links up to and including all plant to landfill links
NANDPM	-	total number of links up to and including all plant to market links

8.8.2 Procedure

Each card of the transportation subfile is tested first whether it is an end of section card (ICODE =9) and then whether it is in the correct section. If INREP = 2 then an attempt is made to match IDORIG and IDDEST with the ID of a facility determined by the type of link.

Details of each link are stored in the arrays ARCOST, ORIGID and DESTID. As the cards are read in they are counted by the variables NAND... From these counts the number of entries for each type of link is known for later processing.

8.9 Subroutine TSETUP

TSETUP is called only if the user has set the switch TWRITE to 1 and it has not been set to 0 by the subroutine ERROR. By the time it is called, all data for use by the MIP and RWP are stored in arrays. TSETUP writes TFILE for entry to the MIP. The format of TFILE is as defined by APEX-III or MPSX-MIP. The required format differs only in the method of specifying binary variables.

To properly understand what is done in TSETUP the reader will need to refer to the formulation of the model in Appendix 1 and the reference manual for APEX-III (1) or MPSX (5). Appendix 4 lists a TFILE written for the sample problem reported in Appendix 5 and 6.

8.9.1 Glossary of Important Variables

Most of the variables used in TSETUP are passed through COMMON from earlier routines. The only important variables local to TSETUP are the parts of the row names and variable (column) names arbitrarily made up for use in the model. They correspond to the names used in the formulation in Appendix I. All the names are initialized by DATA statements at the beginning of the subroutine.

8.9.2 Procedure

The very first item on TFILE is the name of the problem (or dataset). The 'ROWS' section provides a name for every row in the model and types it as an equality, inequality, or non-constraining.

The COLUMNS section of TFILE names each variable in the problem, giving its coefficient in every equation where it appears. Most of the variables are material flows in the system, and these variables are written first using information from COMMON/LNKDAT/. Next there is a variable for each facility throughput. The final part of the COLUMNS section introduces the integer variables named DELTAS. The integer variables are set off with MARKER cards, if TFILE is being written for MPSX-MIP but these are not necessary for APEX-III.

The RHS section gives values to the right-hand-side of equations in the model. If the right-hand-side is zero, this need not be explicitly specified. In this model, the only non-zero right-hand-side values are for amount of waste at each source, and market location demands.

In the BOUNDS section, the integer DELTAS are given an upper bound of 0, and a status of FX if the facility must not be used, an upper bound of 1 and a status of FX if the facility must be used. If the model is deciding the status as part of the analysis, the bound value is 1 and the status is UP for MPSX-MIP or BV for APEX-III. The last record of TFILE is ENDATA.

8.10 Subroutine DSETUP

DSETUP is called only if the user has set the switch DWRITE to 1 and it has not been turned off by subroutine ERROR. The routine writes to DFILE from COMMON all those counts, ID numbers and location names required by the RWP.

8.11 Subroutine REGRES

This routine determines linear regression coefficients from facility capital and operating costs. All variables are passed via the argument list.

8.11.1 Glossary of Important Variables

N - number of observations
X - vector of observations
Y - vector of matching y observations
F - value of y intercept ('fixed' cost)
V - value of slope of line ('variable' cost)
RSQ - measure of goodness of fit

8.11.2 Calculations

If N = 1

If there is just one observation, no regression is possible.

Then $F = Y(1)$
$V = 0$
$RSQ = 0$

If N greater than 1

Step 1 Calculate

$$SX = \Sigma X_i$$

$$SXX = \Sigma (X_i)^2$$

$$SY = \Sigma Y_i$$

$$SYY = \Sigma (Y_i)^2$$

$$SXY = \Sigma X_i Y_i$$

and

$$VARNX = SXX - SX * SX/N$$

$$VARNY = SYY - SY * SY/N$$

$$VARNXY = SXY - SX * SY/N$$

Step 2 Using the model $Y = F + VX$ estimate V and F by

$$V = VARNYX/VARNX$$

and

$$F = SY/N - V * SX/N$$

If F equals 0 continue with Step 3 or else go to step 4.

Step 3 Calculate the RSQ "goodness of fit" measure by

$$RSQ = V * V * VARNX/VARNY$$

and exit the subroutine.

Step 4 Since the model in Step 2 gave a negative value to F, this can be a problem. Instead the model Y = VX is used.

Then F = 0

Calculate V = SXY/SXX

RSQ = V*SXY/SYY

8.12 FUNCTION USPWF

This function computes the uniform series present worth factor to use for calculating the present discounted value of a uniform stream of annual payments.

8.12.1 Glossary of Important Variables

R - discount rate fraction
N - number of years of payments

8.12.2 Calculations

On January 1st of year 1, payment of \$1 on December 31st of years 1, 2, 3 ... n has discounted value:

$$\frac{(1 + R)^n - 1}{R\,(1+R)^n} \qquad \text{if R not equal to 0}$$

If R = 0 (no discounting), then the stream of \$1 payments has value \$N.

CHAPTER NINE

DOCUMENTATION OF THE REPORT WRITER PROGRAM

This chapter gives details of the RWP routine by routine (Figure 15). There are two Versions of the Program, one for use on CDC with APEX-III, the other for use on an IBM/370 using MPSX-MIP. Apart from trivial differences between CDC and IBM FORTRAN, the only important difference in the two versions is in the routine TFETCH which reads all required information on the solution found by the MIP and stores it in arrays in COMMON.

9.1 Main Program

In the main program, subroutines are called to read first DFILE, then TFILE, into core storage. Routines are then called to produce each report in turn.

9.2 Subroutine DFETCH

This routine reads into COMMON storage all the information written on DFILE. Each READ statement in DFETCH matches a WRITE statement in DWRITE of ESP.

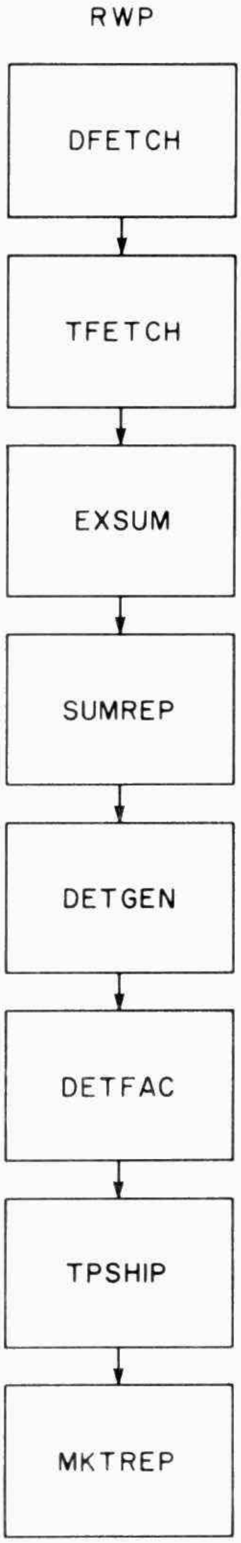
9.3 Subroutine TFETCH for MPSX-MIP

This routine is a modification of the FORTRAN program on p. 235 of the MPSX Manual (5). All information required by RWP from the solution is read from the output file and stored in arrays.

9.3.1 Glossary of Important Variables

ARCWST	-	an array of material flows along each link
WST	-	an array of material flow through each facility
WM	-	an array of material shipments to each market location
DELTA	-	an array containing the value of the integer variable corresponding to each facility
COST	-	an array containing total capital, transportation cost etc.
VALUES	-	an array containing the optimal values of all model variables

FIGURE 15 - SEQUENCE FOR EXECUTION OF SUBROUTINES IN RWP



9.3.2 Procedure

The identification array of the solution file prepared by MPSX-MIP is read and gives the number of rows and columns in the output data. Only the values of the rows near the end of the rows section are required, those giving total capital cost, total facility operating cost, total transportation cost, and total revenue. This information, originally in elements of the array VALUES, is stored in the array COST.

The value of every variable in the columns section is required by RWP. After completing reading of the rows section of the solution file, elements of the array VALUES are stored in the array ARCWST, one for each transportation link. Following this the array VALUES gives material flows through facilities. These are stored in the array WST, first for transfer stations, then for plants, then for landfills. After this comes total material received at each market location stored in the array WM. Finally the array VALUES contains the optimal values of the integer variables, one for each facility. They are stored in the array DELTA first for transfer stations, then for plants, and finally for landfills.

9.4 Subroutine TFETCH for APEX-III

Details of the FORTRAN-readable solution file created by APEX-III are given in Appendix B of the manual (1). All information required by RWP from the solution is read sequentially from the output file and stored in arrays in COMMON.

9.4.1 Glossary of Important Variables

ARCWST - an array of material flows along each link

WST - an array of material flow through each facility

WM - an array of material shipments to each market location

DELTA - an array containing the value of the integer variable corresponding to each facility

COST - an array containing total capital, transportation cost etc.

W - each record of the file output by APEX-III is made up of seven words, arbitrarily named W. The second word, W(2) contains the numeric values of the row and column variables.

9.4.2 Procedure

The first two records of the APEX-III output file are header records. The second of these contains, in words W(6) and W(7) the number of rows and columns respectively. Only the values of the rows near the end of the rows section are required, those giving total capital cost, total facility operating cost, total transportation cost, and total revenue. This information originally in the elements W(2), is stored in the array COST.

The values of every variable in the columns section is required by RWP. After completing reading of the rows section of the solution file, the columns section is read. For each record, the value in W(2) is stored in an array. First come the amounts of flow along links. This information is stored in the array ARCWST, one for each transportation link. Following this are the material flows through facilities. These are stored in the array WST, first for transfer stations, then for plants, then for landfills. After this comes total material received at each market location, stored in the array WM. Last come the optimal values of the integer variables, one for each facility. The values in W(2) are stored in the array DELTA, first for transfer stations, then for plants and finally for landfills.

9.5 Subroutine EXSUM

This routine prints a summary of the cost figures for the solution formed by MIP, the status of all facilities in the system (used-Yes or No) and whether any of the market locations is supplied with recovered material. Some of the figures are available immediately in arrays in COMMON/OUT/ while others must be divided by certain array elements

9.5.1 Glossary of Important Variables

ALWAST	-	total waste generated (passed from ESP)
TNAM	-	name of a transfer station
TUSE	-	status of a transfer station
PNAM	-	name of a plant
PUSE	-	status of a plant
LNAM	-	name of a landfill
LUSE	-	status of a landfill
MNAM	-	name of a market location

MUSE - whether or not there are shipments to and market location
CAP( ) - total capital cost for a type of facility
OP( ) - total operating cost for a type of facility
TR( ) - total transportation cost for shipments from each type of facility

9.5.2 Procedure

First the various totals in COST are converted to a cost per year and a cost per tonne as well as a percentage of total cost. The amount of material recovered is determined by a summation. These various figures are printed as the 'System Financial Summary' and 'Material Flow in the System'.

For the section 'Facilities and Markets Used' the status of each facility is determined by the values in the array DELTA (broken up into DT, DR and DM). A market is 'Used' if shipments to it (WM) are greater than zero.

The other part of the report is a summary of capital, operating, and transportation costs by type of facility. Capital and operating costs are determined by summation over all facilities.

FCP * DELTA + VCP * WST for capital
and FOP * DELTA + VOP * WST for operating
where the fixed and variable capital costs (FCP and VCP) and fixed and variable operating costs (FOP and VOP) are passed from ESP and the values of DELTA and WST (throughput) come from the output of the MIP. Transport costs are derived by summing the cost ARCOST * ARCWST over each link leaving a facility.

9.6 Subroutine SUMREP

This routine produces a report of shipments and cost of shipments from each source to the facility at which the collector truck disposes of the waste.

9.6.1 Glossary of Important Variables

NANDSL - total number of links from sources to transfer stations, plants and landfills
SRCIDN - ID number for a source
ORIGID - ID number for the origin of a link
ARCWST - material shipment along a link
ARCOST - cost of moving a tonne of material along a link

9.6.2 Procedure

For each source, all links from that source are scanned to determine the shipment of materials along them. The scanning is carried out by attempting to match the ORIGID of a link to the SRCIDN of the source being considered. Where there is a match, a check is made whether the flow in the link, ARCWST is greater than zero. If so, the flow is multiplied by ARCOST to give the total cost of shipping along that link. Averages per year and per tonne are calculated.

9.7 Subroutine DETGEN

This routine produces a report of total and weighted average cost of shipping waste from each source. If required, total and weighted average costs can be calculated for groups of sources.

9.7.1 Glossary of Important Variables

NOSUBT - number of groups of sources for which subtotals are required
IDSUBT - an array of ID numbers of the last source in each group requiring subtotals
NANDSL - total number of links from sources to transfer stations, plants and landfills
SRCIDN - ID number for a source
ORIGID - ID number for the origin of a link
ARCWST - material shipment along a link
ARCOST - cost of moving a tonne of material along a link

9.7.2 Procedure

For each source, all links from that source are scanned to determine the shipment of material along them. The scanning is carried out by attempting to match the ORIGID of a link to the SRCIDN of the source being considered. Where there is a match, the material flows on the link ARCWST, and the total cost of shipping it ARCWST * ARCOST, are accumulated. When the scanning for links from one source is completed, averages are calculated and results for that source printed out. The process is continued until results have been calculated and printed for each source.

If subtotals are requested, a similar procedure is used, accumulating not for just one source, but for each source up to and including the one with the IDSUBT being considered.

9.8 Subroutine DETFAC

This routine reports for each facility (transfer station, plant, landfill) the throughput of the facility (waste shipped to it) the capital and operating cost, and the total cost of transporting material from it.

9.8.1 Glossary of Important Variables

NUM - array containing number of transfer stations, plants, landfills, market locations
NAM - array of facility names
IDN - array of facility ID numbers
FCP - array of facility fixed capital costs
VCP - array of facility variable capital costs
FOP - array of facility fixed operating costs
VOP - array of facility variable operating costs
WST - array of facility throughputs
ARCWST - material shipment along a link
DELTA - integer variable for the status of a facility
ARCOST - cost of moving a tonne of material along a link
ORIGID - the ID number for the origin of a link

9.8.2 Procedure

The whole routine is a DO loop iterated once for each facility. For an intermediate facility (transfer station or plant) the cost of transportation from that facility is found by scanning all links from that facility and accumulating the cost of shipping material from it. Facility capital cost is calculated from the facility fixed and variable cost, DELTA value and throughput WST. Similarly for operating cost. Finally a cost per tonne is calculated and all information for the facility printed before calculations begin for the next facility.

9.9 Subroutine MKTREP

If the modelling includes the option of resource recovery, this subroutine is called to report any shipments to market. The Marketing Report lists shipments to each market by product and prints the revenue received.

9.9.1 Glossary of Important Variables

NRESRC - the number of different resources
NUMMKT - the number of markets for each resource
WM - total resource shipped to each market location
RSCNAM - name of a resource
MKTNAM - name of a market location

9.9.2 Procedure

For each resource, and each market location for each resource, a check is made whether the total shipment to it, WM, is greater than zero. If so, the name, amount shipped, and revenue are printed.

9.10 Subroutine TPSHIP

This routine prints a report of shipments from each intermediate facility (transfer station or plant) in this system. The routine is very similar to SUMREP which produces the report of shipments from sources.

9.10.1 Glossary of Important Variables

NNDSL1 -	Position in the arrays of link information of the first transfer station to plant link
NANDTP -	position in the arrays of link information of the last transfer station to plant link
NANDTL -	position in the arrays of link information of the last transfer station to landfill link
NANDPL -	position in the arrays of link information of the last plant to landfill link
NANDPM -	position in the arrays of link information of the last plant to market link
TRFIDN -	transfer station ID number
TRFNAM -	transfer station name
PLTIDN -	plant ID number
PLTNAM -	plant name
ORIGID -	ID number for the origin of a link
DESTID -	ID number for the destination of a link
ARCWST -	material shipped along a link
ARCOST -	cost of moving a tonne of material along a link

9.10.2 Procedure

Transfer stations are considered first. Each link leaving a transfer station, recognized by ORIGID matching TRFIDN, is checked to see if there is material flow along it. If so the amount of waste is noted, the cost of shipping is calculated, a cost per tonne and a cost per year calculated, and these figures printed out.

After considering all transfer stations, results are found for the plants in a similar way and the results printed. Shipments from plants can be to landfills or market locations.

REFERENCES

1. Control Data Corporation, *APEX-III Reference Manual*, *Version 1.2*, CDC Data Services Publications Minneapolis, Minn., 1979.

2. Control Data Corporation, *Cybernet Services SCOPE 3.4 Reference Manual*, CDC Data Services Publication, Minneapolis, Minn., 1977.

3. Control Data Corporation, *Cybernet Services NOS Computing Service Reference Set*, CDC Data Services Publications, Minneapolis, Minn., 1978.

4. Hillier F.S. and G.J. Lieberman, *Introduction to Operations Research*, 3rd ed., Holden-Day Inc., San Francisco, Calif., 1980.

5. International Business Machines Corporation, *Mathematical Programming System Extended / 370 (MPSX/370)* Program Reference Manual, IBM Technical Publications Department, White Plains, N.Y.

6. International Business Machines Corporation, *Mixed Integer Programming / 370 (MIP/370) Program Reference Manual,* IBM Technical Publications Department, White Plains, N.Y.

7. Jenkins, A.L., *Optimal Location of Facilities for Recycling Municipal Solid Waste in Southern Ontario*, Ph.D. thesis, Department of Industrial Engineering, University of Toronto, January, 1979.

8. Jenkins, A.L., *A Comparative Evaluation of Waste Management Systems Models*, Ontario Ministry of the Environment, October, 1977.

9. Wagner, H.M., *Principles of Operations Research* 2nd ed., Prentice-Hall Inc., Englewood Cliffs, N.J., 1975.

APPENDIX 1

MATHEMATICAL FORMULATION OF THE MODEL

APPENDIX I

FORMULATION OF THE MODEL

The notation given here is that used in subroutine TSETUP of the Edit and Setup Program (ESP). The indices used are the identity numbers of the sources, facilities, resources and markets. Symbolism of the form $\Sigma_t ST_{st}$ means include in the equation every ST variable with index s and arbitrary t.

Indices

s - indixes the sources
t - indexes the transfer stations
p - indexes the plants
l - indexes the landfills
r - indexes the materials recoverable
m - indexes the market locations

Parameters

$SRCTOT_s$ - quantity of waste generated by source s in the period

$TRFLLT_t$ - lower limit on throughput of transfer station t in the period (if used)
$TRFULT_t$ - upper limit on throughput of transfer station t in the period (if used)
$TRFFCP_t$ - fixed capital cost of transfer station t
$TRFVCP_t$ - variable capital cost of transfer station t
$TRFFOP_t$ - fixed operating cost of transfer station t
$TRFVOP_t$ - variable operating cost of transfer station t

$PLTLLT_p$ - lower limit on throughput of plant p in the period (if used)

$PLTULT_p$ - upper limit on throughput of plant p in the period (if used)

$PLTFCP_p$ - fixed capital cost of plant p

$PLTVCP_p$ - variable capital cost of plant p

$PLTFOP_p$ - fixed operating cost of plant p

$PLTVOP_p$ - variable operating cost of plant p

$PLTKEP_p$ - proportion of waste entering plant p and leaving it as resources or waste residue

$PLTRRN_{pr}$ - proportion of waste entering plant p which may be recovered as resource r

$LFLLLT_l$ - lower limit on throughput of landfill l in the period (if used)

$LFLULT_l$ - upper limit on throughput of landfill l in the period (if used)

$LFLFCP_l$ - fixed capital cost of landfill l

$LFLVCP_l$ - variable cost of landfill l

$LFLFOP_l$ - fixed operating cost of landfill l

$LFLVOP_l$ - variable operating cost of landfill l

$DEMAND_{rm}$ - greatest amount of resource r that market m will buy

$PRICET_{rm}$ - price per tonne payable for resource r at market m

$ARCOST_{od}$ - cost of moving a tonne of material from origin o to destination d

LTOLFL - limit on total tonnes of waste landfilled in the system over the period

LNUMTR - limit on the number of transfer stations in the system

LNUMPL - limit on the number of plants in the system

LNUMPL - limit on the number of landfills in the system

LCAPTL - limit on total capital expenditure to set up the system

Decision Variables

Most of the decision variables are the material flows along links. One decision variable is set up for each link defined in the transportation link subfile. Other variables are for flow of material through facilities - the W. . . variables, and the integer decision variables for each facility D . . .

ST_{st} - flow from source s to transfer station t

SP_{sp} - flow from source s to plant p

SL_{sl} - flow from source s to landfill l

TP_{tp} - flow from transfer station t to plant p

TL_{tl} - flow from transfer station t to landfill l

PL_{pl} - flow from plant p to landfill l

PM_{prm} - flow from plant p of resource r to market m

$WTRF_{t}$ - flow through transfer station t

$WPLT_{p}$ - flow through plant p

$WLFL_{l}$ - flow to landfill l

$WMKT_{rm}$ - flow of resource r to market location m

$DTRF_{t}$ - integer variable

= 0 if transfer station t not used

= 1 if transfer station t used

$DPLT_{p}$ - integer variable

= 0 if plant p not used

= 1 if plant p used

$DLFL_{l}$ - integer variable

= 0 if landfill l not used

- 1 if landfill l used

Row Names

These are made up as shown in the listing of equations, e.g., there is one row $SRCGN_s$ for each source.

Constraints

ROW NAME	MEANING
$SRCGN_s$	Total shipments from source s to transfer stations, plants and landfills equals the waste generation at s. $\Sigma_t ST_{st} + \Sigma_p SP_{sp} + \Sigma_l SL_{sl} = SRCTOT_s$
$TFRTO_t$	Total shipments from sources to transfer station t equals the station throughput. $\Sigma_t ST_{st} - WTRF_t = 0$
$TRLES_t$	Throughput of transfer station t is greater than the lower limit on throughput, if the station is used. $- WTRF_t + TRFLLT_t DTRF_t \leqslant 0$
$TRMOS_t$	Throughput of transfer station t is less than the upper limit on throughput, if the station is used at all. $- WTRF_t + TRFULT_t DTRF_t \geqslant 0$
$TRFRM_t$	Throughput of transfer station t equals all shipments from it to plants and landfills. $- WTRF_t + {}_p TP_{tp} + {}_l TL_{tl} = 0$

ROW NAME	MEANING
$PLNTO_p$	Total shipments from sources and transfer stations to plant t equals the plant throughput. $\Sigma_s SP_{sp} + \Sigma_t TP_{tp} - WPLT_p = 0$
$PLEST_p$	Throughput of plant p is greater than the lower limit on throughput, if the plant is used. $- WPLT_p + PLTLLT_p DPLT_p \leqslant 0$
$PLMOS_p$	Throughput of plant p is less than the upper limit on throughput, if the plant is used at all. $- WPLT_p + PLTULT_p DPLT_p \geqslant 0$
$PLFRM_p$	A proportion of the throughput of plant p equals all shipments from it to landfills and markets. $- PLTKEP_p WPLT_p + \Sigma_l PL_{pl} + \Sigma_r (\Sigma_m PM_{prm}) = 0$
$PLAR_{pr}$	Shipments from plant p of resource r cannot exceed a specified proportion of the plant throughput. $- PLTRRN_{pr} WPLT_p + \Sigma_m PM_{prm} \leqslant 0$ for all r

ROW NAME	MEANING
$LFLTO_l$	Total shipments from sources, transfer stations and plants to landfill l equals the landfill throughput. $\Sigma_s SL_{sl} + \Sigma_t TL_{tl} + \Sigma_p PL_{pl} - WLFL_l = 0$
$LFLES_l$	Throughput of landfill l is greater than the lower limit on throughput, if the landfill is used. $- WLFL_l + LFLLLT_l DLFL_l \leqslant 0$
$LFMOS_l$	Throughput of landfill l is less than the upper limit on throughput, if the landfill is used at all. $- WLFL_l + LFLULT_l DLFL_l \geqslant 0$
$MAKTO_{rm}$	Total shipments from all plants of resource r to market location m equals the amount received. $\Sigma_p PM_{prm} - WMKT_{rm} = 0$
$MKMOS_{rm}$	The amount of resource r received at market location m cannot exceed the demand. $WMKT_{rm} \leqslant DEMAND_{rm}$

ROW NAME	MEANING
LFLTOTAL	Total throughput of all landfills must be less than a specified limit. $\Sigma_l WLFL_l \leqslant LTOLFL$
NUMTRANF	The total number of transfer stations opened must not exceed a specified limit. $\Sigma_t DTRF_t \leqslant LNUMTR$
NUMPLANT	The total number of plants opened must not exceed a specified limit. $\Sigma_p DPLT_p \leqslant LNUMPL$
NUMLNDFL	The total number of landfills opened must not exceed a specified limit. $\Sigma_l DLFL_l \leqslant LNUMLF$
CAPITALC	Total capital expenditure is made up of fixed and variable costs for transfer stations, plants, and landfills. It must not exceed a specified limit. $\Sigma_t (TRFFCP_t DTRF_t + TRFVCP_t WTRF_t)$ $+ \Sigma_p (PLTFCP_p DPLT_p + PLTVCP_p WPLT_p)$ $+ \Sigma_l (LFLFCP_l DLFL_l + LFLVCP_l WLFL_l) \leqslant LCAPTL$

ROW NAME	MEANING
OPERATNG	Total operating cost is made up of fixed and variable costs for transfer stations, plants and landfills. $\Sigma_t (TRFFOP_t + TRFVOP_t WTRF_t)$ $+ \Sigma_p (PLTFOP_p DPLT_p + PLTVOP_p WPLT_p)$ $+ \Sigma_l (LFLFOP_l DLFL_l + LFLVOP_p WLFL_l)$
TRANSPRT	Total transport cost is made up of all costs for shipments from sources to transfer stations, sources to plants, sources to landfills, transfer stations to plants, transfer stations to landfills, plants to landfills, and plants to markets. $\Sigma_{j \epsilon \text{ st arcs}} ST_j ARCOST_j + \Sigma_{j \epsilon \text{ sp arcs}} SP_j ARCOST_j$ $+ \Sigma_{j \epsilon \text{ sl arcs}} SL_j ARCOST_j + \Sigma_{j \epsilon \text{ tp arcs}} TP_j ARCOST_j$ $+ \Sigma_{j \epsilon \text{ tl arcs}} TL_j ARCOST_j + \Sigma_{j \epsilon \text{ pl arcs}} PL_j ARCOST_j$ $+ \Sigma_{j \epsilon \text{ pm arcs}} PM_j ARCOST_j$
REVENUES	Total revenue is made up of the amount received at each market location, times the price, added up for all resources and all markets. $\Sigma_r (\Sigma_m WMKT_m PRICET_{rm})$
NETCOST	CAPITALC + OPERATNG + TRANSPRT - REVENUES

APPENDIX 2

SAMPLE DATA INPUT FILE

```
      LITTLE  KINGSTON-QUINTE
       TEST DATA
 1981 2000       3.0    2    2    1    1    1
    2    6    3    3    6    3
    2    1    1    0
         0         0
         5         5         5
   PACKER   2.0   LONGHAL  0.3
121SIDNEY     1
121           400000
125TRENTON    1
125           600
126BELLEVL    1
126          170000
141NAPANEE    1
141           500000
152KINGENV    1
152           600000
154KINGSTON   4
154          .8    0     61360   .8
221BELLEVIL   1               1000    100000    1    3    3
221      1     37500      1981    590000
221      1     75000      1981   1185000
221      1    200000      1981   2375000
221           37500    100000
221           75000    150000
221          200000    320000
233NAPANEF    0              10000    120000    1    3    3
233      1     37500      1991    610000
233      1     80000      1987   1100000
233      1    190000      1990   2040000
233           37500    100000
233           75000    170000
233          200000    280000
243KINGSTON   2                        200000    1    4    2
243      1     30000      1980    700000
243      1     80000      2001    900000
243      1    190000      1980   2100000
243      1    210000      1985   2200000
243           75000    170000
243          200000    280000
321BELLVILE   2               1000    100000    1    3    3
321      1     37500      1981   4290000
321      1     75000      1981   8480000
321      1    200000      1981  12960000
321           37500    453000
321           75000    695000
321          200000   1040000
321            5000  500  200
333NAPAN      2              0     50000    1    1    3
333      1                1991   7000000
333           40000    400000
333           50000    500000
333           20000    100000
333            5000  500  200
343KINGSTON   2              0    200000    1    3    3
343      1     40000      1982   4300000
343      1     80000      1983   8500000
343      1    200000      1984  13000000
343           37500    500000
343           75000    700000
343          190000    905000
343         500 5000  600  200
419SIDNEY-I   2              0    100000    1    3    3   2000000
```

```
419      1    125000      1981    235500
419      1    187500      1981    268500
419      1    250000      1981    352500
419           125000     80000
419           187500     87000
419           250000    111000
422TYENDING   2              0    100000    0    1    2   1800000
422                       2000     30000
422           100000     70000
422           200000    100000
432RICHMOND   1          10000    100000    0    1    1   1000000
432                       1990     50000
432                     100000
434ERNESTWN   0              0     80000    1    2    1   1000000
434      1    100000      1990    300000
434      1    200000      1990    500000
434           187500     87000
441STORRNT    2              0    100000    1    2    3   2000000
441      1    100000      1990    245000
441      1    200000      1985    300000
441           100000     80000
441           150000     90000
441           200000    190000
444KINGSTON   2              0    100000    1    3    3   2000000
444      1    120000      1981    220000
444      1    180000      1981    270000
444      1    260000      1981    360000
444           124000     80000
444           200000     90000
444           260000    110000
R.D.F.
156PICTON                25000    100000
157BATH                  25000    120000
FERROUS
258HAMILTON              50000     10000
GLASS
339TORONTO               50000     10000
1 121  221    1   28
1 125  221    1   44
1 126  221    1    5
1 126  233    1   57
1 141  221    1   70
1 141  233    1    3
1 141  243    1   45
1 152  233    1   46
1 152  243    1    5
1 154  233    1   53
1 154  243    1    7
9
2 121  321    1   14
2 121  333    1   50
2 125  321    1   29
2 125  333    1   70
2 126  343    1  101
2 126  321    2    4
2 126  333    2   25
2 141  321    2   84
2 141  333    1   70
2 141  343    2    3
2 152  321    2   45
2 152  333    1   43
2 152  343    2   44
2 154  321    2    5
2 154  333    1   90
2 154  343    2   50
9
```

```
3 121   419     2    12
3 121   422     2    30
3 121   432     2    50
3 125   419     1    14
3 125   422     1    42
3 125   432     2    64
3 126   419     1    21
3 126   422     2    22
3 126   432     2    40
3 141   422     1    58
3 141   434     1     7
3 141   444     1    19
3 152   432     2    26
3 152   444     2    28
3 152   441     2    32
3 154   432     1    34
3 154   434     1    10
3 154   441     1    62
3 154   444     1     8
9
4 221   333     2    44
4 221   343     2    86
4 233   321     2    40
4 233   343     2    45
4 243   321     2    87
4 243   333     2    22
9
5 221   419     2    12
5 221   422     2    16
5 221   432     2    37
5 233   419     2    80
5 233   422     2    56
5 233   432     2    12
5 233   434     2    21
5 233   441     2    87
5 233   444     2    50
5 243   422     2    72
5 243   432     2    44
5 243   434     2    48
5 243   441     2    17
5 243   444     2     8
9
6 321   419     2    12
6 321   422     2    17
6 321   432     2    36
6 333   419     2    50
6 333   422     2    28
6 333   432     2    16
6 333   434     2    20
6 333   441     2    80
6 333   444     2    50
6 343   422     2    72
6 343   432     2    55
6 343   434     2    47
6 343   441     2    25
6 343   444     2     8
9
 7 321   156     2    25
 7 321   157     2    45
 7 321   258     2     0
 7 321   339     2   100
 7 333   156     2    35
 7 333   157     2    75
 7 333   258     2     0
 7 333   339     2    25
 7 343   156     2    27
```

```
7 343  157    2   33
7 343  258    2    0
7 343  339    2  121
9
```

APPENDIX 3

OUTPUT FROM ESP

```
********************************************************************************************* PAGE NO.  1
*                                                                                           * 80/08/08.
*     ONTARIO MINISTRY OF THE ENVIRONMENT -- WASTE MANAGEMENT SYSTEMS MODEL-VERSION 1.1 -- EDIT AND SETUP PROGRAM     *
*                                                                                           *
*         LITTLE  KINGSTON-QUINTE                    TEST DATA                     OUTPUT FROM ROUTINE START     *
*                                                                                           *
*********************************************************************************************

        STUDY PERIOD 1981-2000(20 YEARS)
        DISCOUNT RATE IS  3.00 %


                         GIVEN OR      MAXIMUM
                         POSSIBLE      ALLOWED
                         LOCATIONS     IN RUN                  OPTIONS SELECTED IN THE RUN

       SOURCES               6                                 INPUT REPORT  - COMPLETE
       TRANSFER STATIONS     3            5                    OUTPUT REPORT - COMPLETE
       PLANTS                3            5                    LINEAR APPROXIMATION OF LANDFILL COSTS
       LANDFILLS             6            5                    WRITE FILE T
                                                               WRITE FILE D


       LIMITS OVER THE PERIOD FOR:
           TOTAL LANDFILLING        999999999. TONNES
           CAPITAL EXPENDITURES $ 999999999.


        RESOURCE       NUMBER OF
                       MARKETS

            1             2
            2             1
            3             1


        VEHICLE DATA

           VEHICLE         NAME            COST PER TONNE
           NUMBER                          PER HOUR, $

              1          PACKER               2.000
              2          LONGHAL               .300
```

```
*********************************************************************************************** PAGE NO.   2
*                                                                                             * 80/06/08.
*    ONTARIO MINISTRY OF THE ENVIRONMENT -- WASTE MANAGEMENT SYSTEMS MODEL-VERSION 1.1 -- EDIT AND SETUP PROGRAM   *
*                                                                                             *
*         LITTLE  KINGSTON-QUINTE                 TEST DATA                  OUTPUT FROM ROUTINE SOURCE     *
*                                                                                             *
***********************************************************************************************

SOURCES - DATA TYPE 1
                                                                                              TOTAL
                      TOTAL                                                                   WASTE
        SOURCE        WASTE                                                                   TONNES
LABEL    NAME         TONNES
                                                                                             400000.
 121   SIDNEY         400000.                                                                600000.
 125   TRENTON        600000.                                                               1700000.
 126   BELLEVL       1700000.                                                                500000.
 141   NAPANEE        500000.                                                                600000.
 152   KINGENV        600000.

----------------------------------------------------------------------------------------------


SOURCES - DATA TYPE 4

                      WASTE          WASTE                          POPULATION                TOTAL
         SOURCE     FIRST YEAR       GROWTH        POPULATION        GROWTH                   WASTE
LABEL     NAME    TONNES/CAPITA    % PER YEAR      FIRST YEAR     % PER YEAR                  TONNES

 154   KINGSTON        .800            0.00           61360.          .80                   1060080.

----------------------------------------------------------------------------------------------



                 TOTAL WASTE FOR THE AREA AND FOR THE WHOLE PERIOD, TONNES-------------------------------      4860080.2
```

```
************************************************************************************************************************ PAGE NO.   3
*                                                                                                                      * 80/08/08.
*     ONTARIO MINISTRY OF THE ENVIRONMENT -- WASTE MANAGEMENT SYSTEMS MODEL-VERSION 1.1 -- EDIT AND SETUP PROGRAM      *
*                                                                                                                      *
*           LITTLE  KINGSTON-QUINTE                  TEST DATA                          OUTPUT FROM ROUTINE TRANFR     *
*                                                                                                                      *
************************************************************************************************************************


TRANSFER STATIONS WITH LUMP PAYMENTS OF CAPITAL COST


                                             ANNUAL                                                   ANNUAL    ANNUAL      ANNUAL
                        THROLGHPUT,TONNES  CAPACITY         LUMP PAYMENTS OF CAPITAL COST            CAPITAL  THROUGHPUT  OPERATING
LABEL  NAME    STATUS     LEAST      MOST    TONNES   YEAR  -   $    YEAR  -   $    YEAR  -   $       COST,$    TONNES      COST,$

 221 BELLEVIL   USED        1000.   100000.    37500. 1981   590000.    0        0.    0        0.    590000.     37500.    100000.
 221                           0.        0.    75000. 1981  1185000.    0        0.    0        0.   1185000.     75000.    150000.
 221                           0.        0.   200000. 1981  2375000.    0        0.    0        0.   2375000.    200000.    320000.

 233 NAPANEE  NOT USED     10000.   120000.    37500. 1991   610000.    0        0.    0        0.    453897.     37500.    100000.
 233                           0.        0.    80000. 1987  1100000.    0        0.    0        0.    921233.     80000.    170000.
 233                           0.        0.   190000. 1990  2040000.    0        0.    0        0.   1563490.    190000.    280000.

 243 KINGSTON   MODEL          0.   200000.    30000. 1980   700000.    0        0.    0        0.    721000.     30000.    170000.
 243                           0.        0.    80000. 2001   900000.    0        0.    0        0.    498308.     80000.    280000.
 243                           0.        0.   190000. 1980  2100000.    0        0.    0        0.   2163000.         0.         0.
 243                           0.        0.   210000. 1985  2200000.    0        0.    0        0.   1954672.         0.         0.


TRANSFER STATIONS - RESULTS OF REGRESSION

LABEL   NAME        INTERCEPT      SLOPE      RSQUARE   CAPITAL COST VS. CAPACITY   OPERATIONAL COST VS.THROUGHPUT

 221  BELLEVIL         48812.9     1.35540     1.000              X
 221  BELLEVIL        277518.0    10.61583      .988                                          X
 233  NAPANEE          73920.9     1.05036      .970              X
 233  NAPANEE         263193.3     6.98885      .975                                          X
 243  KINGSTON        104000.0      .88000     1.000              X
 243  KINGSTON        185577.6     9.00916      .850                                          X
```

```
********************************************************************************************************************** PAGE NO.  4
*                                                                                                                    * 80/08/08.
*     ONTARIO MINISTRY OF THE ENVIRONMENT -- WASTE MANAGEMENT SYSTEMS MODEL-VERSION 1.1 -- EDIT AND SETUP PROGRAM    *
*                                                                                                                    *
*         LITTLE  KINGSTON-QUINTE                        TEST DATA                        OUTPUT FROM ROUTINE PLANTS    *
*                                                                                                                    *
**********************************************************************************************************************



PLANTS WITH LUMP PAYMENTS OF CAPITAL COST


                                                  ANNUAL                                                             ANNUAL     ANNUAL      ANNUAL
                             THROUGHPUT,TONNES    CAPACITY        LUMP PAYMENTS OF CAPITAL COST                      CAPITAL   THROUGHPUT  OPERATING
LABEL  NAME       STATUS       LEAST      MOST    TONNES    YEAR  -   $     YEAR  -   $     YEAR  -   $             COST,$     TONNES      COST,$

 321 BELLVILE     MODEL        1000.   100000.     37500. 1981  4290000.    0          0.    0          0.   4290000.     37500.     453000.
 321                             0.        0.      75000. 1981  8480000.    0          0.    0          0.   8480000.     75000.     695000.
 321                             0.        0.     200000. 1981 12960000.    0          0.    0          0.  12960000.    200000.    1040000.


 333 NAPAN        MODEL          0.    50000.         0. 1991  7000000.    0          0.    0          0.   5208657.         0.     400000.
 333                             0.        0.         0.    0         0.    0          0.    0          0.         0.         0.     500000.
 333                             0.        0.         0.    0         0.    0          0.    0          0.         0.         0.     100000.


 343 KINGSTON     MODEL          0.   200000.     40000. 1982  4300000.    0          0.    0          0.   4174757.     40000.     500000.
 343                             0.        0.      80000. 1983  8500000.    0          0.    0          0.   8012065.     80000.     700000.
 343                             0.        0.     200000. 1984 13000000.    0          0.    0          0.  11896842.    200000.     905000.



       PLANT      % WASTE  ---------------% WASTE RECOVERED AS RESOURCE---------------
LABEL  NAME        LOST       1       2       3       4       5       6       7       8       9

 321  BELLVILE     0.00    50.00    5.00    2.00    0.00    0.00    0.00    0.00    0.00    0.00
 333  NAPAN        0.00    50.00    5.00    2.00    0.00    0.00    0.00    0.00    0.00    0.00
 343  KINGSTON     5.00    50.00    6.00    2.00    0.00    0.00    0.00    0.00    0.00    0.00


PLANTS - RESULTS OF REGRESSION

LABEL    NAME            INTERCEPT         SLOPE         RSQUARE   CAPITAL COST VS. CAPACITY   OPERATIONAL COST VS.THROUGHPUT

 321  BELLVILE           375406.5        3.39770         .960                X
 321  BELLVILE          3478345.3       48.94388         .923                                              X
 333  NAPAN                   0.0        9.55556         .978                X
 343  KINGSTON           454590.1        2.45035         .925                X
 343  KINGSTON          3270967.6       44.59613         .925                                              X
```

```
********************************************************************************************************** PAGE NO.  5
*                                                                                                        * 80/08/08.
*     ONTARIO MINISTRY OF THE ENVIRONMENT -- WASTE MANAGEMENT SYSTEMS MODEL-VERSION 1.1 -- EDIT AND SETUP PROGRAM     *
*                                                                                                        *
*         LITTLE  KINGSTON-QUINTE                  TEST DATA                   OUTPUT FROM ROUTINE LANDFL  *
*                                                                                                        *
**********************************************************************************************************

LANDFILLS WITH AMORTIZED CAPITAL COST

                                                      ANNUAL     AMORTIZATION          ANNUAL       ANNUAL       ANNUAL
                                THROUGHPUT LIMITS,TONNES  CAPACITY   LAST    ANNUAL          CAPITAL     THROUGHPUT    OPERATING
LABEL     NAME       STATUS        LEAST          MOST      TONNES     YEAR  PAYMENT,$         COST,$       TONNES        COST,$

 422   TYENDING     MODEL             0.      100000.          0.   2000     30000.        459714.      100000.       70000.
 422                                  0.          0.           0.      0         0.             0.      200000.      100000.

 432   RICHMOND     USED          10000.      100000.          0.   1990     50000.        439305.           0.      100000.


LANDFILLS WITH LUMP PAYMENTS OF CAPITAL COST

                                              ANNUAL                                                    ANNUAL     ANNUAL       ANNUAL
                          THROUGHPUT,TONNES  CAPACITY          LUMP PAYMENTS OF CAPITAL COST            CAPITAL   THROUGHPUT   OPERATING
LABEL  NAME      STATUS      LEAST    MOST     TONNES   YEAR  -   $     YEAR  -   $     YEAR  -   $     COST,$     TONNES       COST,$

 419 SIDNEY-I    MODEL          0.   100000.   125000. 1981   235500.    0        0.    0         0.   235500.    125000.      80000.
 419                            0.        0.   187500. 1981   268500.    0        0.    0         0.   268500.    187500.      87000.
 419                            0.        0.   250000. 1981   352500.    0        0.    0         0.   352500.    250000.     111000.

 434 ERNESTWN NOT USED          0.    80000.   100000. 1990   300000.    0        0.    0         0.   229925.    100000.      87000.
 434                            0.        0.   200000. 1990   500000.    0        0.    0         0.   383208.         0.          0.

 441 STORRNT     MODEL          0.   100000.   100000. 1990   245000.    0        0.    0         0.   187772.    100000.      80000.
 441                            0.        0.   200000. 1985   300000.    0        0.    0         0.   266546.    200000.      90000.
 441                            0.        0.        0.    0        0.    0        0.    0         0.        0.         0.     190000.

 444 KINGSTON    MODEL          0.   100000.   120000. 1981   220000.    0        0.    0         0.   220000.    120000.      80000.
 444                            0.        0.   180000. 1981   270000.    0        0.    0         0.   270000.    180000.      90000.
 444                            0.        0.   260000. 1981   360000.    0        0.    0         0.   360000.    260000.     110000.
```

```
LANDFILLS - RESULTS OF REGRESSION

LABEL   NAME          INTERCEPT       SLOPE       RSQUARE   CAPITAL COST VS. CAPACITY   OPERATIONAL COST VS.THROUGHPUT

 419  SIDNEY-I         46166.7       .24800        .909                x
 419  SIDNEY-I        110000.0       .93600        .949                                           x
 422  TYENDING         40000.0       .30000       1.000                x
 434  ERNESTWN         76641.7      1.53283       1.000                                           x
 441  STORRNT              0.0       .82069        .965                x
 441  STORRNT         108998.1       .78774       1.000                                           x
 444  KINGSTON         51148.1       .21670        .935                x
 444  KINGSTON         95405.4      1.00676        .993                                           x
```

```
***************************************************************************************************************************** PAGE NO.  7
*                                                                                                                           * 80/08/08.
*     ONTARIO MINISTRY OF THE ENVIRONMENT -- WASTE MANAGEMENT SYSTEMS MODEL-VERSION 1.1 -- EDIT AND SETUP PROGRAM           *
*                                                                                                                           *
*           LITTLE  KINGSTON-QUINTE                          TEST DATA                            OUTPUT FROM ROUTINE MARKET *
*                                                                                                                           *
*****************************************************************************************************************************



                                                                 ANNUAL        TOTAL
  RESOURCE           MARKET  MARKET             PRICE            DEMAND        DEMAND
NUMBER   NAME        LABEL    NAME             $/TONNE           TONNES        TONNES

   1     R.D.F.       156    PICTON              25.00          100000.      2000000.
   1     R.D.F.       157    BATH                25.00          120000.      2400000.

   2     FERROUS      258    HAMILTON            50.00           10000.       200000.

   3     GLASS        339    TORONTO             50.00           10000.       200000.
```

```
****************************************************************************************************** PAGE NO.  8
*                                                                                                    * 80/08/08.
*     ONTARIO MINISTRY OF THE ENVIRONMENT -- WASTE MANAGEMENT SYSTEMS MODEL-VERSION 1.1 -- EDIT AND SETUP PROGRAM   *
*                                                                                                    *
*          LITTLE  KINGSTON-QUINTE                  TEST DATA                    OUTPUT FROM ROUTINE LINKEM      *
*                                                                                                    *
******************************************************************************************************



                                                                  ROUND      COST, $
  -------ORIGIN-------      -----DESTINATION----- VEHICLE          TRIP      PER TONNE
 LABEL    NAME      TYPE     LABEL    NAME     TYPE     TYPE          MIN       PER TRIP

   121  SIDNEY       SR        221  BELLEVIL    TR     PACKER         28.0        .69428
   125  TRENTON      SR        221  BELLEVIL    TR     PACKER         44.0       1.09101
   126  BELLEVL      SR        221  BELLEVIL    TR     PACKER          5.0        .12398
   126  BELLEVL      SR        233  NAPANEE     TR     PACKER         57.0       1.41336
   141  NAPANEE      SR        221  BELLEVIL    TR     PACKER         70.0       1.73571
   141  NAPANEE      SR        233  NAPANEE     TR     PACKER          3.0        .07439
   141  NAPANEE      SR        243  KINGSTON    TR     PACKER         45.0       1.11581
   152  KINGENV      SR        233  NAPANEE     TR     PACKER         46.0       1.14061
   152  KINGENV      SR        243  KINGSTON    TR     PACKER          5.0        .12398
   154  KINGSTON     SR        233  NAPANEE     TR     PACKER         53.0       1.31418
   154  KINGSTON     SR        243  KINGSTON    TR     PACKER          7.0        .17357

   121  SIDNEY       SR        321  BELLVILE    PL     PACKER         14.0        .34714
   121  SIDNEY       SR        333  NAPAN       PL     PACKER         50.0       1.23979
   125  TRENTON      SR        321  BELLVILE    PL     PACKER         29.0        .71908
   125  TRENTON      SR        333  NAPAN       PL     PACKER         70.0       1.73571
   126  BELLEVL      SR        343  KINGSTON    PL     PACKER        101.0       2.50437
   126  BELLEVL      SR        321  BELLVILE    PL     LONGHAL         4.0        .01488
   126  BELLEVL      SR        333  NAPAN       PL     LONGHAL        25.0        .09298
   141  NAPANEE      SR        321  BELLVILE    PL     LONGHAL        84.0        .31243
   141  NAPANEE      SR        333  NAPAN       PL     PACKER         70.0       1.73571
   141  NAPANEE      SR        343  KINGSTON    PL     LONGHAL         3.0        .01116
   152  KINGENV      SR        321  BELLVILE    PL     LONGHAL        45.0        .16737
   152  KINGENV      SR        333  NAPAN       PL     PACKER         43.0       1.06622
   152  KINGENV      SR        343  KINGSTON    PL     LONGHAL        44.0        .16365
   154  KINGSTON     SR        321  BELLVILE    PL     LONGHAL         5.0        .01860
   154  KINGSTON     SR        333  NAPAN       PL     PACKER         90.0       2.23162
   154  KINGSTON     SR        343  KINGSTON    PL     LONGHAL        50.0        .18597

   121  SIDNEY       SR        419  SIDNEY-I    LF     LONGHAL        12.0        .04463
   121  SIDNEY       SR        422  TYENDING    LF     LONGHAL        30.0        .11158
   121  SIDNEY       SR        432  RICHMOND    LF     LONGHAL        50.0        .18597
   125  TRENTON      SR        419  SIDNEY-I    LF     PACKER         14.0        .34714
   125  TRENTON      SR        422  TYENDING    LF     PACKER         42.0       1.04142
   125  TRENTON      SR        432  RICHMOND    LF     LONGHAL        64.0        .23804
   126  BELLEVL      SR        419  SIDNEY-I    LF     PACKER         21.0        .52071
   126  BELLEVL      SR        422  TYENDING    LF     LONGHAL        22.0        .08183
   126  BELLEVL      SR        432  RICHMOND    LF     LONGHAL        40.0        .14877
   141  NAPANEE      SR        422  TYENDING    LF     PACKER         58.0       1.43816
   141  NAPANEE      SR        434  ERNESTWN    LF     PACKER          7.0        .17357
   141  NAPANEE      SR        444  KINGSTON    LF     PACKER         19.0        .47112
   152  KINGENV      SR        432  RICHMOND    LF     LONGHAL        26.0        .09670
   152  KINGENV      SR        444  KINGSTON    LF     LONGHAL        28.0        .10414
   152  KINGENV      SR        441  STOKRNT     LF     LONGHAL        32.0        .11902
```

```
****************************************************************************************************** PAGE NO.  9
*                                                                                                    * 80/08/08.
*     ONTARIO MINISTRY OF THE ENVIRONMENT -- WASTE MANAGEMENT SYSTEMS MODEL-VERSION 1.1 -- EDIT AND SETUP PROGRAM     *
*                                                                                                    *
*          LITTLE  KINGSTON-QUINTE                    TEST DATA                  OUTPUT FROM ROUTINE LINKEM     *
*                                                                                                    *
******************************************************************************************************


                                                                     ROUND      COST, $
  -------ORIGIN-------       -----DESTINATION-----  VEHICLE        TRIP      PER TONNE
LABEL   NAME      TYPE     LABEL   NAME      TYPE    TYPE          MIN      PER TRIP

  154  KINGSTON    SR        432  RICHMOND   LF     PACKER        34.0       .84306
  154  KINGSTON    SR        434  ERNESTWN   LF     PACKER        10.0       .24796
  154  KINGSTON    SR        441  STORRNT    LF     PACKER        62.0      1.53734
  154  KINGSTON    SR        444  KINGSTON   LF     PACKER         8.0       .19837

  221  BELLEVIL    TR        333  NAPAN      PL     LONGHAL       44.0       .16365
  221  BELLEVIL    TR        343  KINGSTON   PL     LONGHAL       86.0       .31987
  233  NAPANEE     TR        321  BELLVILE   PL     LONGHAL       40.0       .14877
  233  NAPANEE     TR        343  KINGSTON   PL     LONGHAL       45.0       .16737
  243  KINGSTON    TR        321  BELLVILE   PL     LONGHAL       87.0       .32359
  243  KINGSTON    TR        333  NAPAN      PL     LONGHAL       22.0       .08183

  221  BELLEVIL    TR        419  SIDNEY-I   LF     LONGHAL       12.0       .04463
  221  BELLEVIL    TR        422  TYENDING   LF     LONGHAL       16.0       .05951
  221  BELLEVIL    TR        432  RICHMOND   LF     LONGHAL       37.0       .13762
  233  NAPANEE     TR        419  SIDNEY-I   LF     LONGHAL       80.0       .29755
  233  NAPANEE     TR        422  TYENDING   LF     LONGHAL       56.0       .20828
  233  NAPANEE     TR        432  RICHMOND   LF     LONGHAL       12.0       .04463
  233  NAPANEE     TR        434  ERNESTWN   LF     LONGHAL       21.0       .07811
  233  NAPANEE     TR        441  STORRNT    LF     LONGHAL       87.0       .32359
  233  NAPANEE     TR        444  KINGSTON   LF     LONGHAL       50.0       .18597
  243  KINGSTON    TR        422  TYENDING   LF     LONGHAL       72.0       .26779
  243  KINGSTON    TR        432  RICHMOND   LF     LONGHAL       44.0       .16365
  243  KINGSTON    TR        434  ERNESTWN   LF     LONGHAL       48.0       .17853
  243  KINGSTON    TR        441  STORRNT    LF     LONGHAL       17.0       .06323
  243  KINGSTON    TR        444  KINGSTON   LF     LONGHAL        8.0       .02975

  321  BELLVILE    PL        419  SIDNEY-I   LF     LONGHAL       12.0       .04463
  321  BELLVILE    PL        422  TYENDING   LF     LONGHAL       17.0       .06323
  321  BELLVILE    PL        432  RICHMOND   LF     LONGHAL       36.0       .13390
  333  NAPAN       PL        419  SIDNEY-I   LF     LONGHAL       50.0       .18597
  333  NAPAN       PL        422  TYENDING   LF     LONGHAL       28.0       .10414
  333  NAPAN       PL        432  RICHMOND   LF     LONGHAL       16.0       .05951
  333  NAPAN       PL        434  ERNESTWN   LF     LONGHAL       20.0       .07439
  333  NAPAN       PL        441  STORRNT    LF     LONGHAL       80.0       .29755
  333  NAPAN       PL        444  KINGSTON   LF     LONGHAL       50.0       .18597
  343  KINGSTON    PL        422  TYENDING   LF     LONGHAL       72.0       .26779
  343  KINGSTON    PL        432  RICHMOND   LF     LONGHAL       55.0       .20457
  343  KINGSTON    PL        434  ERNESTWN   LF     LONGHAL       47.0       .17481
  343  KINGSTON    PL        441  STORRNT    LF     LONGHAL       25.0       .09298
  343  KINGSTON    PL        444  KINGSTON   LF     LONGHAL        8.0       .02975

  321  BELLVILE    PL        156  PICTON     MR     LONGHAL       25.0       .09298
  321  BELLVILE    PL        157  BATH       MR     LONGHAL       45.0       .16737
```

```
**************************************************************************************************** PAGE NO. 10
*                                                                                                  * 80/03/08.
*     ONTARIO MINISTRY OF THE ENVIRONMENT -- WASTE MANAGEMENT SYSTEMS MODEL-VERSION 1.1 -- EDIT AND SETUP PROGRAM     *
*                                                                                                  *
*          LITTLE  KINGSTON-QUINTE                    TEST DATA                    OUTPUT FROM ROUTINE LINKEM     *
*                                                                                                  *
****************************************************************************************************



                                                                          ROUND      COST, $
       --------ORIGIN-------      -----DESTINATION----- VEHICLE           TRIP       PER TUNNE
       LABEL   NAME       TYPE    LABEL   NAME       TYPE    TYPE          MIN        PER TRIP

        321  BELLVILE     PL        258  HAMILTON    MR    LONGHAL       0.0        0.00000
        321  BELLVILE     PL        339  TORONTO     MR    LONGHAL     100.0         .37194
        333  NAPAN        PL        156  PICTON      MR    LONGHAL      35.0         .13018
        333  NAPAN        PL        157  BATH        MR    LONGHAL      75.0         .27895
        333  NAPAN        PL        258  HAMILTON    MR    LONGHAL       0.0        0.00000
        333  NAPAN        PL        339  TORONTO     MR    LONGHAL      25.0         .09298
        343  KINGSTON     PL        156  PICTON      MR    LONGHAL      27.0         .10042
        343  KINGSTON     PL        157  BATH        MR    LONGHAL      33.0         .12274
        343  KINGSTON     PL        258  HAMILTON    MR    LONGHAL       0.0        0.00000
        343  KINGSTON     PL        339  TORONTO     MR    LONGHAL     121.0         .45004

                 END OF TRANSPORTATION ROUTES.
```

```
***************************************************************************************************************************** PAGE NO. 11
*                                                                                                                           * 80/08/08.
*     ONTARIO MINISTRY OF THE ENVIRONMENT -- WASTE MANAGEMENT SYSTEMS MODEL-VERSION 1.1 -- EDIT AND SETUP PROGRAM         *
*                                                                                                                           *
*          LITTLE  KINGSTON-QUINTE                    TEST DATA                          OUTPUT FROM ROUTINE TSETUP         *
*                                                                                                                           *
*****************************************************************************************************************************

     WRITE TO T FILE COMPLETED.
```

```
********************************************************************************************************** PAGE NO. 12
*                                                                                                        * 80/08/08.
*     ONTARIO MINISTRY OF THE ENVIRONMENT -- WASTE MANAGEMENT SYSTEMS MODEL-VERSION 1.1 -- EDIT AND SETUP PROGRAM     *
*                                                                                                        *
*          LITTLE  KINGSTON-QUINTE                    TEST DATA                 OUTPUT FROM ROUTINE DSETUP   *
*                                                                                                        *
**********************************************************************************************************

 WRITE OF FILE D COMPLETED.
```

APPENDIX 4

INTERMEDIATE TFILE

```
NAME          ONTWAST
ROWS
 E  SRCGN121
 E  SRCGN125
 E  SRCGN126
 E  SRCGN141
 E  SRCGN152
 E  SRCGN154
 E  TRFTO221
 L  TRLES221
 G  TRMOS221
 E  TRFRM221
 E  TRFTO233
 L  TRLES233
 G  TRMOS233
 E  TRFRM233
 E  TRFTO243
 L  TRLES243
 G  TRMOS243
 E  TRFRM243
 E  PLNTO321
 L  PLEST321
 G  PLMOS321
 E  PLFRM321
 L  PLAR3211
 L  PLAR3212
 L  PLAR3213
 E  PLNTO333
 L  PLEST333
 G  PLMOS333
 E  PLFRM333
 L  PLAR3331
 L  PLAR3332
 L  PLAR3333
 E  PLNTO343
 L  PLEST343
 G  PLMOS343
 E  PLFRM343
 L  PLAR3431
 L  PLAR3432
 L  PLAR3433
 E  LFLTO419
 L  LFLES419
 G  LFMOS419
 E  LFLTO422
 L  LFLES422
 G  LFMOS422
 E  LFLTO432
 L  LFLES432
 G  LFMOS432
 E  LFLTO434
 L  LFLES434
 G  LFMOS434
 E  LFLTO441
 L  LFLES441
 G  LFMOS441
 E  LFLTO444
 L  LFLES444
 G  LFMOS444
 E  MAKTO156
 L  MKMOS156
 E  MAKTO157
 L  MKMOS157
 E  MAKTO258
 L  MKMOS258
 E  MAKT0339
 L  MKMOS339
 L  LFLTOTAL
 L  NUMTRANF
 L  NUMPLANT
 L  NUMLNDFL
 L  CAPITALC
 N  OPERATNG
 N  TRANSPRT
 N  REVENUES
 N  NETCOST
 N  CHCAPITL
 N  CHOPRTNG
 N  CHTRNPRT
 N  CHREVENU
COLUMNS
    ST121221  SRCGN121        1.000000
    ST121221  TRFT0221        1.000000
    ST121221  TRANSPRT         .694282
    ST121221  NETCOST          .694282
    ST121221  CHTRNPRT         .694282
    ST125221  SRCGN125        1.000000
    ST125221  TRFT0221        1.000000
    ST125221  TRANSPRT        1.091015
    ST125221  NETCOST         1.091015
    ST125221  CHTRNPRT        1.091015
    ST126221  SRCGN126        1.000000
    ST126221  TRFT0221        1.000000
    ST126221  TRANSPRT         .123979
    ST126221  NETCOST          .123979
    ST126221  CHTRNPRT         .123979
    ST126233  SRCGN126        1.000000
    ST126233  TRFT0233        1.000000
    ST126233  TRANSPRT        1.413360
    ST126233  NETCOST         1.413360
    ST126233  CHTRNPRT        1.413360
    ST141221  SRCGN141        1.000000
    ST141221  TRFT0221        1.000000
    ST141221  TRANSPRT        1.735705
    ST141221  NETCOST         1.735705
    ST141221  CHTRNPRT        1.735705
    ST141233  SRCGN141        1.000000
    ST141233  TRFT0233        1.000000
    ST141233  TRANSPRT         .074387
    ST141233  NETCOST          .074387
    ST141233  CHTRNPRT         .074387
    ST141243  SRCGN141        1.000000
    ST141243  TRFT0243        1.000000
    ST141243  TRANSPRT        1.115811
    ST141243  NETCOST         1.115811
    ST141243  CHTRNPRT        1.115811
    ST152233  SRCGN152        1.000000
    ST152233  TRFT0233        1.000000
    ST152233  TRANSPRT        1.140606
    ST152233  NETCOST         1.140606
    ST152233  CHTRNPRT        1.140606
    ST152243  SRCGN152        1.000000
    ST152243  TRFT0243        1.000000
    ST152243  TRANSPRT         .123979
    ST152243  NETCOST          .123979
    ST152243  CHTRNPRT         .123979
    ST154233  SRCGN154        1.000000
    ST154233  TRFT0233        1.000000
    ST154233  TRANSPRT        1.314177
    ST154233  NETCOST         1.314177
```

```
ST154233   CHTRNPRT        1.314177
ST154243   SRCGN154        1.000000
ST154243   TRFT0243        1.000000
ST154243   TRANSPRT         .173571
ST154243   NETCOST          .173571
ST154243   CHTRNPRT         .173571
SP121321   SRCGN121        1.000000
SP121321   PLNT0321        1.000000
SP121321   TRANSPRT         .347141
SP121321   NETCOST          .347141
SP121321   CHTRNPRT         .347141
SP121333   SRCGN121        1.000000
SP121333   PLNT0333        1.000000
SP121333   TRANSPRT        1.239790
SP121333   NETCOST         1.239790
SP121333   CHTRNPRT        1.239790
SP125321   SRCGN125        1.000000
SP125321   PLNT0321        1.000000
SP125321   TRANSPRT         .719078
SP125321   NETCOST          .719078
SP125321   CHTRNPRT         .719078
SP125333   SRCGN125        1.000000
SP125333   PLNT0333        1.000000
SP125333   TRANSPRT        1.735705
SP125333   NETCOST         1.735705
SP125333   CHTRNPRT        1.735705
SP126343   SRCGN126        1.000000
SP126343   PLNT0343        1.000000
SP126343   TRANSPRT        2.504375
SP126343   NETCOST         2.504375
SP126343   CHTRNPRT        2.504375
SP126321   SRCGN126        1.000000
SP126321   PLNT0321        1.000000
SP126321   TRANSPRT         .014877
SP126321   NETCOST          .014877
SP126321   CHTRNPRT         .014877
SP126333   SRCGN126        1.000000
SP126333   PLNT0333        1.000000
SP126333   TRANSPRT         .092984
SP126333   NETCOST          .092984
SP126333   CHTRNPRT         .092984
SP141321   SRCGN141        1.000000
SP141321   PLNT0321        1.000000
SP141321   TRANSPRT         .312427
SP141321   NETCOST          .312427
SP141321   CHTRNPRT         .312427
SP141333   SRCGN141        1.000000
SP141333   PLNT0333        1.000000
SP141333   TRANSPRT        1.735705
SP141333   NETCOST         1.735705
SP141333   CHTRNPRT        1.735705
SP141343   SRCGN141        1.000000
SP141343   PLNT0343        1.000000
SP141343   TRANSPRT         .011158
SP141343   NETCOST          .011158
SP141343   CHTRNPRT         .011158
SP152321   SRCGN152        1.000000
SP152321   PLNT0321        1.000000
SP152321   TRANSPRT         .167372
SP152321   NETCOST          .167372
SP152321   CHTRNPRT         .167372
SP152333   SRCGN152        1.000000
SP152333   PLNT0333        1.000000
SP152333   TRANSPRT        1.066219
SP152333   NETCOST         1.066219
SP152333   CHTRNPRT        1.066219
```

```
SP152343   SRCGN152          1.000000
SP152343   PLNT0343          1.000000
SP152343   TRANSPRT           .163652
SP152343   NETCOST            .163652
SP152343   CHTRNPRT           .163652
SP154321   SRCGN154          1.000000
SP154321   PLNT0321          1.000000
SP154321   TRANSPRT           .018597
SP154321   NETCOST            .018597
SP154321   CHTRNPRT           .018597
SP154333   SRCGN154          1.000000
SP154333   PLNT0333          1.000000
SP154333   TRANSPRT          2.231621
SP154333   NETCOST           2.231621
SP154333   CHTRNPRT          2.231621
SP154343   SRCGN154          1.000000
SP154343   PLNT0343          1.000000
SP154343   TRANSPRT           .185968
SP154343   NETCOST            .185968
SP154343   CHTRNPRT           .185968
SL121419   SRCGN121          1.000000
SL121419   LFLT0419          1.000000
SL121419   TRANSPRT           .044632
SL121419   NETCOST            .044632
SL121419   CHTRNPRT           .044632
SL121422   SRCGN121          1.000000
SL121422   LFLT0422          1.000000
SL121422   TRANSPRT           .111581
SL121422   NETCOST            .111581
SL121422   CHTRNPRT           .111581
SL121432   SRCGN121          1.000000
SL121432   LFLT0432          1.000000
SL121432   TRANSPRT           .185968
SL121432   NETCOST            .185968
SL121432   CHTRNPRT           .185968
SL125419   SRCGN125          1.000000
SL125419   LFLT0419          1.000000
SL125419   TRANSPRT           .347141
SL125419   NETCOST            .347141
SL125419   CHTRNPRT           .347141
SL125422   SRCGN125          1.000000
SL125422   LFLT0422          1.000000
SL125422   TRANSPRT          1.041423
SL125422   NETCOST           1.041423
SL125422   CHTRNPRT          1.041423
SL125432   SRCGN125          1.000000
SL125432   LFLT0432          1.000000
SL125432   TRANSPRT           .238040
SL125432   NETCOST            .238040
SL125432   CHTRNPRT           .238040
SL125419   SRCGN126          1.000000
SL126419   LFLT0419          1.000000
SL126419   TRANSPRT           .520712
SL126419   NETCOST            .520712
SL126419   CHTRNPRT           .520712
SL126422   SRCGN126          1.000000
SL126422   LFLT0422          1.000000
SL126422   TRANSPRT           .081826
SL126422   NETCOST            .081826
SL126422   CHTRNPRT           .081826
SL126432   SRCGN126          1.000000
SL126432   LFLT0432          1.000000
SL126432   TRANSPRT           .148775
SL126432   NETCOST            .148775
SL126432   CHTRNPRT           .148775
SL141422   SRCGN141          1.000000
```

```
SL141422  LFLT0422          1.000000
SL141422  TRANSPRT          1.438156
SL141422  NETCOST           1.438156
SL141422  CHTRNPRT          1.438156
SL141434  SRCGN141          1.000000
SL141434  LFLT0434          1.000000
SL141434  TRANSPRT           .173571
SL141434  NETCOST            .173571
SL141434  CHTRNPRT           .173571
SL141444  SRCGN141          1.000000
SL141444  LFLT0444          1.000000
SL141444  TRANSPRT           .471120
SL141444  NETCOST            .471120
SL141444  CHTRNPRT           .471120
SL152432  SRCGN152          1.000000
SL152432  LFLT0432          1.000000
SL152432  TRANSPRT           .096704
SL152432  NETCOST            .096704
SL152432  CHTRNPRT           .096704
SL152444  SRCGN152          1.000000
SL152444  LFLT0444          1.000000
SL152444  TRANSPRT           .104142
SL152444  NETCOST            .104142
SL152444  CHTRNPRT           .104142
SL152441  SRCGN152          1.000000
SL152441  LFLT0441          1.000000
SL152441  TRANSPRT           .119020
SL152441  NETCOST            .119020
SL152441  CHTRNPRT           .119020
SL154432  SRCGN154          1.000000
SL154432  LFLT0432          1.000000
SL154432  TRANSPRT           .843057
SL154432  NETCOST            .843057
SL154432  CHTRNPRT           .843057
SL154434  SRCGN154          1.000000
SL154434  LFLT0434          1.000000
SL154434  TRANSPRT           .247958
SL154434  NETCOST            .247958
SL154434  CHTRNPRT           .247958
SL154441  SRCGN154          1.000000
SL154441  LFLT0441          1.000000
SL154441  TRANSPRT          1.537339
SL154441  NETCOST           1.537339
SL154441  CHTRNPRT          1.537339
SL154444  SRCGN154          1.000000
SL154444  LFLT0444          1.000000
SL154444  TRANSPRT           .198365
SL154444  NETCOST            .198366
SL154444  CHTRNPRT           .198366
TP221333  TRFRM221          1.000000
TP221333  PLNT0333          1.000000
TP221333  TRANSPRT           .163652
TP221333  NETCOST            .163652
TP221333  CHTRNPRT           .163652
TP221343  TRFRM221          1.000000
TP221343  PLNT0343          1.000000
TP221343  TRANSPRT           .319866
TP221343  NETCOST            .319866
TP221343  CHTRNPRT           .319866
TP233321  TRFRM233          1.000000
TP233321  PLNT0321          1.000000
TP233321  TRANSPRT           .148775
TP233321  NETCOST            .148775
TP233321  CHTRNPRT           .148775
TP233343  TRFRM233          1.000000
TP233343  PLNT0343          1.000000
```

```
TP233343   TRANSPRT        .167372
TP233343   NETCOST         .167372
TP233343   CHTRNPRT        .167372
TP243321   TRFRM243       1.000000
TP243321   PLNT0321       1.000000
TP243321   TRANSPRT        .323585
TP243321   NETCOST         .323585
TP243321   CHTRNPRT        .323585
TP243333   TRFRM243       1.000000
TP243333   PLNT0333       1.000000
TP243333   TRANSPRT        .081826
TP243333   NETCOST         .081826
TP243333   CHTRNPRT        .081826
TL221419   TRFRM221       1.000000
TL221419   LFLT0419       1.000000
TL221419   TRANSPRT        .044632
TL221419   NETCOST         .044632
TL221419   CHTRNPRT        .044632
TL221422   TRFRM221       1.000000
TL221422   LFLT0422       1.000000
TL221422   TRANSPRT        .059510
TL221422   NETCOST         .059510
TL221422   CHTRNPRT        .059510
TL221432   TRFRM221       1.000000
TL221432   LFLT0432       1.000000
TL221432   TRANSPRT        .137617
TL221432   NETCOST         .137617
TL221432   CHTRNPRT        .137617
TL233419   TRFRM233       1.000000
TL233419   LFLT0419       1.000000
TL233419   TRANSPRT        .297549
TL233419   NETCOST         .297549
TL233419   CHTRNPRT        .297549
TL233422   TRFRM233       1.000000
TL233422   LFLT0422       1.000000
TL233422   TRANSPRT        .208285
TL233422   NETCOST         .208285
TL233422   CHTRNPRT        .208285
TL233432   TRFRM233       1.000000
TL233432   LFLT0432       1.000000
TL233432   TRANSPRT        .044632
TL233432   NETCOST         .044632
TL233432   CHTRNPRT        .044632
TL233434   TRFRM233       1.000000
TL233434   LFLT0434       1.000000
TL233434   TRANSPRT        .078107
TL233434   NETCOST         .078107
TL233434   CHTRNPRT        .078107
TL233441   TRFRM233       1.000000
TL233441   LFLT0441       1.000000
TL233441   TRANSPRT        .323585
TL233441   NETCOST         .323585
TL233441   CHTRNPRT        .323585
TL233444   TRFRM233       1.000000
TL233444   LFLT0444       1.000000
TL233444   TRANSPRT        .185968
TL233444   NETCOST         .185968
TL233444   CHTRNPRT        .185968
TL243422   TRFRM243       1.000000
TL243422   LFLT0422       1.000000
TL243422   TRANSPRT        .267795
TL243422   NETCOST         .267795
TL243422   CHTRNPRT        .267795
TL243432   TRFRM243       1.000000
TL243432   LFLT0432       1.000000
TL243432   TRANSPRT        .163652
```

```
TL243432  NETCOST       .163552
TL243432  CHTRNPRT      .163552
TL243434  TRFRM243     1.000000
TL243434  LFLT0434     1.000000
TL243434  TRANSPRT      .178530
TL243434  NETCOST       .178530
TL243434  CHTRNPRT      .178530
TL243441  TRFRM243     1.000000
TL243441  LFLT0441     1.000000
TL243441  TRANSPRT      .063229
TL243441  NETCOST       .063229
TL243441  CHTRNPRT      .063229
TL243444  TRFRM243     1.000000
TL243444  LFLT0444     1.000000
TL243444  TRANSPRT      .029755
TL243444  NETCOST       .029755
TL243444  CHTRNPRT      .029755
PL321419  PLFRM321     1.000000
PL321419  LFLT0419     1.000000
PL321419  TRANSPRT      .044632
PL321419  NETCOST       .044632
PL321419  CHTRNPRT      .044632
PL321422  PLFRM321     1.000000
PL321422  LFLT0422     1.000000
PL321422  TRANSPRT      .063229
PL321422  NETCOST       .063229
PL321422  CHTRNPRT      .063229
PL321432  PLFRM321     1.000000
PL321432  LFLT0432     1.000000
PL321432  TRANSPRT      .133897
PL321432  NETCOST       .133897
PL321432  CHTRNPRT      .133897
PL333419  PLFRM333     1.000000
PL333419  LFLT0419     1.000000
PL333419  TRANSPRT      .185968
PL333419  NETCOST       .185968
PL333419  CHTRNPRT      .185968
PL333422  PLFRM333     1.000000
PL333422  LFLT0422     1.000000
PL333422  TRANSPRT      .104142
PL333422  NETCOST       .104142
PL333422  CHTRNPRT      .104142
PL333432  PLFRM333     1.000000
PL333432  LFLT0432     1.000000
PL333432  TRANSPRT      .059510
PL333432  NETCOST       .059510
PL333432  CHTRNPRT      .059510
PL333434  PLFRM333     1.000000
PL333434  LFLT0434     1.000000
PL333434  TRANSPRT      .074387
PL333434  NETCOST       .074387
PL333434  CHTRNPRT      .074387
PL333441  PLFRM333     1.000000
PL333441  LFLT0441     1.000000
PL333441  TRANSPRT      .297549
PL333441  NETCOST       .297549
PL333441  CHTRNPRT      .297549
PL333444  PLFRM333     1.000000
PL333444  LFLT0444     1.000000
PL333444  TRANSPRT      .185968
PL333444  NETCOST       .185968
PL333444  CHTRNPRT      .185968
PL343422  PLFRM343     1.000000
PL343422  LFLT0422     1.000000
PL343422  TRANSPRT      .267795
PL343422  NETCOST       .267795
```

```
PL343422  CHTRNPRT     .267795
PL343432  PLFRM343    1.000000
PL343432  LFLT0432    1.000000
PL343432  TRANSPRT     .204565
PL343432  NETCOST      .204565
PL343432  CHTRNPRT     .204565
PL343434  PLFRM343    1.000000
PL343434  LFLT0434    1.000000
PL343434  TRANSPRT     .174310
PL343434  NETCOST      .174310
PL343434  CHTRNPRT     .174310
PL343441  PLFRM343    1.000000
PL343441  LFLT0441    1.000000
PL343441  TRANSPRT     .092984
PL343441  NETCOST      .092984
PL343441  CHTRNPRT     .092984
PL343444  PLFRM343    1.000000
PL343444  LFLT0444    1.000000
PL343444  TRANSPRT     .029755
PL343444  NETCOST      .029755
PL343444  CHTRNPRT     .029755
PM321156  PLFRM321    1.000000
PM321156  PLAR3211    1.000000
PM321156  MAKT0156    1.000000
PM321156  TRANSPRT     .092984
PM321156  NETCOST      .092984
PM321156  CHTRNPRT     .092984
PM321157  PLFRM321    1.000000
PM321157  PLAR3211    1.000000
PM321157  MAKT0157    1.000000
PM321157  TRANSPRT     .167372
PM321157  NETCOST      .167372
PM321157  CHTRNPRT     .167372
PM321258  PLFRM321    1.000000
PM321258  PLAR3212    1.000000
PM321258  MAKT0258    1.000000
PM321258  TRANSPRT    0.000000
PM321258  NETCOST     0.000000
PM321258  CHTRNPRT    0.000000
PM321339  PLFRM321    1.000000
PM321339  PLAR3213    1.000000
PM321339  MAKT0339    1.000000
PM321339  TRANSPRT     .371937
PM321339  NETCOST      .371937
PM321339  CHTRNPRT     .371937
PM333156  PLFRM333    1.000000
PM333156  PLAR3331    1.000000
PM333156  MAKT0156    1.000000
PM333156  TRANSPRT     .130178
PM333156  NETCOST      .130178
PM333156  CHTRNPRT     .130178
PM333157  PLFRM333    1.000000
PM333157  PLAR3331    1.000000
PM333157  MAKT0157    1.000000
PM333157  TRANSPRT     .278953
PM333157  NETCOST      .278953
PM333157  CHTRNPRT     .278953
PM333258  PLFRM333    1.000000
PM333258  PLAR3332    1.000000
PM333258  MAKT0258    1.000000
PM333258  TRANSPRT    0.000000
PM333258  NETCOST     0.000000
PM333258  CHTRNPRT    0.000000
PM333339  PLFRM333    1.000000
PM333339  PLAR3333    1.000000
PM333339  MAKT0339    1.000000
```

```
PM333339   TRANSPRT      .092984
PM333339   NETCOST       .092984
PM333339   CHTRNPRT      .092984
PM343156   PLFRM343     1.000000
PM343156   PLAR3431     1.000000
PM343156   MAKT0156     1.000000
PM343156   TRANSPRT      .100423
PM343156   NETCOST       .100423
PM343156   CHTRNPRT      .100423
PM343157   PLFRM343     1.000000
PM343157   PLAR3431     1.000000
PM343157   MAKT0157     1.000000
PM343157   TRANSPRT      .122739
PM343157   NETCOST       .122739
PM343157   CHTRNPRT      .122739
PM343258   PLFRM343     1.000000
PM343258   PLAR3432     1.000000
PM343258   MAKT0258     1.000000
PM343258   TRANSPRT     0.000000
PM343258   NETCOST      0.000000
PM343258   CHTRNPRT     0.000000
PM343339   PLFRM343     1.000000
PM343339   PLAR3433     1.000000
PM343339   MAKT0339     1.000000
PM343339   TRANSPRT      .450044
PM343339   NETCOST       .450044
PM343339   CHTRNPRT      .450044
WTRF221    TRFT0221    -1.000000
WTRF221    TRLES221    -1.000000
WTRF221    TRMOS221    -1.000000
WTRF221    TRFRM221    -1.000000
WTRF221    CAPITALC      .530791
WTRF221    OPERATNG     1.008243
WTRF221    NETCOST      1.539035
WTRF233    TRFT0233    -1.000000
WTRF233    TRLES233    -1.000000
WTRF233    TRMOS233    -1.000000
WTRF233    TRFRM233    -1.000000
WTRF233    CAPITALC      .349442
WTRF233    OPERATNG      .781335
WTRF233    NETCOST      1.130777
WTRF243    TRFT0243    -1.000000
WTRF243    TRLES243    -1.000000
WTRF243    TRMOS243    -1.000000
WTRF243    TRFRM243    -1.000000
WTRF243    CAPITALC      .450458
WTRF243    OPERATNG      .654609
WTRF243    NETCOST      1.105067
WPLT321    PLNT0321    -1.000000
WPLT321    PLEST321    -1.000000
WPLT321    PLMOS321    -1.000000
WPLT321    PLFRM321    -1.000000
WPLT321    PLAR3211     -.500000
WPLT321    PLAR3212     -.050000
WPLT321    PLAR3213     -.020000
WPLT321    CAPITALC     2.447194
WPLT321    OPERATNG     2.527458
WPLT321    NETCOST      4.974652
WPLT321    CHCAPITL     2.447194
WPLT321    CHOPRTNG     2.527458
WPLT333    PLNT0333    -1.000000
WPLT333    PLEST333    -1.000000
WPLT333    PLMOS333    -1.000000
WPLT333    PLFRM333    -1.000000
WPLT333    PLAR3331     -.500000
WPLT333    PLAR3332     -.050000
```

```
WPLT333   PLAR3333    -.020000
WPLT333   CAPITALC    0.000000
WPLT333   OPERATNG    7.108127
WPLT333   NETCOST     7.108127
WPLT333   CHCAPITL    0.000000
WPLT333   CHOPRTNG    7.108127
WPLT343   PLNT0343   -1.000000
WPLT343   PLEST343   -1.000000
WPLT343   PLMOS343   -1.000000
WPLT343   PLFRM343    -.950000
WPLT343   PLAR3431    -.500000
WPLT343   PLAR3432    -.060000
WPLT343   PLAR3433    -.020000
WPLT343   CAPITALC    2.229806
WPLT343   OPERATNG    1.822748
WPLT343   NETCOST     4.052555
WPLT343   CHCAPITL    2.229806
WPLT343   CHOPRTNG    1.822748
WLFL419   LFLT0419   -1.000000
WLFL419   LFLES419   -1.000000
WLFL419   LFMOS419   -1.000000
WLFL419   LFLTOTAL    1.000000
WLFL419   CAPITALC     .046800
WLFL419   OPERATNG     .184481
WLFL419   NETCOST      .231281
WLFL422   LFLT0422   -1.000000
WLFL422   LFLES422   -1.000000
WLFL422   LFMOS422   -1.000000
WLFL422   LFLTOTAL    1.000000
WLFL422   CAPITALC    0.000000
WLFL422   OPERATNG     .223162
WLFL422   NETCOST      .223162
WLFL432   LFLT0432   -1.000000
WLFL432   LFLES432   -1.000000
WLFL432   LFMOS432   -1.000000
WLFL432   LFLTOTAL    1.000000
WLFL432   CAPITALC    0.000000
WLFL432   OPERATNG    0.000000
WLFL432   NETCOST     0.000000
WLFL434   LFLT0434   -1.000000
WLFL434   LFLES434   -1.000000
WLFL434   LFMOS434   -1.000000
WLFL434   LFLTOTAL    1.000000
WLFL434   CAPITALC     .076642
WLFL434   OPERATNG    0.000000
WLFL434   NETCOST      .076642
WLFL441   LFLT0441   -1.000000
WLFL441   LFLES441   -1.000000
WLFL441   LFMOS441   -1.000000
WLFL441   LFLTOTAL    1.000000
WLFL441   CAPITALC     .039387
WLFL441   OPERATNG     .610489
WLFL441   NETCOST      .649876
WLFL444   LFLT0444   -1.000000
WLFL444   LFLES444   -1.000000
WLFL444   LFMOS444   -1.000000
WLFL444   LFLTOTAL    1.000000
WLFL444   CAPITALC     .050338
WLFL444   OPERATNG     .161201
WLFL444   NETCOST      .211539
WMKT156   MAKT0156   -1.000000
WMKT156   MKMOS156    1.000000
WMKT156   REVENUES  -18.596844
WMKT156   NETCOST   -18.596844
WMKT156   CHREVENU  -18.596844
WMKT157   MAKT0157   -1.000000
```

```
WMKT157     MKMOS157       1.000000
WMKT157     REVENUES     -18.596844
WMKT157     NETCOST      -18.596844
WMKT157     CHREVENU     -18.596844
WMKT258     MAKT0258      -1.000000
WMKT258     MKMOS258       1.000000
WMKT258     REVENUES     -37.193687
WMKT258     NETCOST      -37.193687
WMKT258     CHREVENU     -37.193687
WMKT339     MAKT0339      -1.000000
WMKT339     MKMOS339       1.000000
WMKT339     REVENUES     -37.193687
WMKT339     NETCOST      -37.193687
WMKT339     CHREVENU     -37.193687
DTRF221     TRLES221        20000.0
DTRF221     TRMOS221      2000000.0
DTRF221     NUMTRANF            1.0
DTRF221     CAPITALC       277518.0
DTRF221     OPERATNG       726213.4
DTRF221     NETCOST       1003731.4
DTRF233     TRLES233       200000.0
DTRF233     TRMOS233      2400000.0
DTRF233     NUMTRANF            1.0
DTRF233     CAPITALC       263183.3
DTRF233     OPERATNG      1099755.8
DTRF233     NETCOST       1362939.1
DTRF243     TRLES243            0.0
DTRF243     TRMOS243      4000000.0
DTRF243     NUMTRANF            1.0
DTRF243     CAPITALC       185577.6
DTRF243     OPERATNG      1547257.4
DTRF243     NETCOST       1732835.0
DPLT321     PLEST321        20000.0
DPLT321     PLMOS321      2000000.0
DPLT321     NUMPLANT            1.0
DPLT321     CAPITALC      3478345.3
DPLT321     OPERATNG      5585100.4
DPLT321     NETCOST       9063445.7
DPLT321     CHCAPITL      3478345.3
DPLT321     CHOPRTNG      5585100.4
DPLT333     PLEST333            0.0
DPLT333     PLMOS333      1000000.0
DPLT333     NUMPLANT            1.0
DPLT333     CAPITALC      5208657.4
DPLT333     OPERATNG            0.0
DPLT333     NETCOST       5208657.4
DPLT333     CHCAPITL      5208657.4
DPLT333     CHOPRTNG            0.0
DPLT343     PLEST343            0.0
DPLT343     PLMOS343      4000000.0
DPLT343     NUMPLANT            1.0
DPLT343     CAPITALC      3270967.6
DPLT343     OPERATNG      6763152.3
DPLT343     NETCOST      10034119.9
DPLT343     CHCAPITL      3270967.6
DPLT343     CHOPRTNG      6763152.3
DLFL419     LFLES419            0.0
DLFL419     LFMOS419      2000000.0
DLFL419     NUMLNDFL            1.0
DLFL419     CAPITALC       110000.0
DLFL419     OPERATNG       686843.4
DLFL419     NETCOST        796843.4
DLFL422     LFLES422            0.0
DLFL422     LFMOS422      1800000.0
DLFL422     NUMLNDFL            1.0
DLFL422     CAPITALC       459714.0
```

```
    DLFL422    OPERATNG        595099.0
    DLFL422    NETCOST        1054813.0
    DLFL432    LFLES432        200000.0
    DLFL432    LFMOS432       1000000.0
    DLFL432    NUMLNDFL             1.0
    DLFL432    CAPITALC        439305.4
    DLFL432    OPERATNG       1487747.5
    DLFL432    NETCOST        1927052.9
    DLFL434    LFLES434             0.0
    DLFL434    LFMOS434       1000000.0
    DLFL434    NUMLNDFL             1.0
    DLFL434    CAPITALC         76641.7
    DLFL434    OPERATNG       1294340.3
    DLFL434    NETCOST        1370982.0
    DLFL441    LFLES441             0.0
    DLFL441    LFMOS441       2000000.0
    DLFL441    NUMLNDFL             1.0
    DLFL441    CAPITALC        108938.1
    DLFL441    OPERATNG             0.0
    DLFL441    NETCOST         108938.1
    DLFL444    LFLES444             0.0
    DLFL444    LFMOS444       2000000.0
    DLFL444    NUMLNDFL             1.0
    DLFL444    CAPITALC         95405.4
    DLFL444    OPERATNG        760954.7
    DLFL444    NETCOST         856360.1
RHS
    ORIGRHS    SRCGN121        400000.0
    ORIGRHS    SRCGN125        600000.0
    ORIGRHS    SRCGN126       1700000.0
    ORIGRHS    SRCGN141        500000.0
    ORIGRHS    SRCGN152        600000.0
    ORIGRHS    SRCGN154       1060080.2
    ORIGRHS    MKMOS156       2000000.0
    ORIGRHS    MKMOS157       2400000.0
    ORIGRHS    MKMOS258        200000.0
    ORIGRHS    MKMOS339        200000.0
    ORIGRHS    LFLTOTAL     999999999.0
    ORIGRHS    NUMTRANF             5.0
    ORIGRHS    NUMPLANT             5.0
    ORIGRHS    NUMLNDFL             5.0
    ORIGRHS    CAPITALC     999999999.0
    CHGGEN     SRCGN121        400000.0
    CHGGEN     SRCGN125        600000.0
    CHGGEN     SRCGN126       1700000.0
    CHGGEN     SRCGN141        500000.0
    CHGGEN     SRCGN152        600000.0
    CHGGEN     SRCGN154       1060080.2
    CHGDMD     MKMOS156       2000000.0
    CHGDMD     MKMOS157       2400000.0
    CHGDMD     MKMOS258        200000.0
    CHGDMD     MKMOS339        200000.0
    CHGTLFL    LFLTOTAL     999999999.0
    CHGCPL     CAPITALC     999999999.0
BOUNDS
 FX FACILITY  DTRF221               1.0
 FX FACILITY  DTRF233               0.0
 UP FACILITY  DTRF243               1.0
 UP FACILITY  DPLT321               1.0
 UP FACILITY  DPLT333               1.0
 UP FACILITY  DPLT343               1.0
 UP FACILITY  DLFL419               1.0
 UP FACILITY  DLFL422               1.0
 FX FACILITY  DLFL432               1.0
 FX FACILITY  DLFL434               0.0
 UP FACILITY  DLFL441               1.0
 UP FACILITY  DLFL444             1.0
ENDATA
```

APPENDIX 5

OUTPUT FROM APEX-111

```
DATE 80/08/08  TIME 15.36.39        A P E X - I I I   C O N T R O L   P R O G R A M        APEX-III 1.202   FIELD LENGTH 200000 OCTAL

     APEX-III CALL CARD - APEX(S,MIN,OBJ=NETCUST,MIP,FS)

                    .LABEL..   .VERB...   .PARAM1.   .PARAM2.   .PARAM3.   .PARAM4.   .PARAM5.

                               INPUT      $
      (GENERATED)              BRANCH     *          MAJERR     MINERR
                               SET        KNDIR      MIN
                               SET        KNOBJ      NETCUST
                               SET        KNRHS      $
                               SET        KNBND      $
                               SET        KNRNG      $
                               SELECT
      (GENERATED)              BRANCH     *          MAJERR
                               CRASH
      (GENERATED)              BRANCH     *          MAJERR
                               PRIMAL
                               BRANCH     PRIMEND    MAJERR     *          PROBOUT
                    PERTURB    SET        RTDELTA    1.0E-5
                    DOPRIML    PRIMAL
                    PRIMEND    BRANCH     *          MAJERR     PROBOUT
                               MIXINT
                               BRANCH     *          MAJERR     PROBSAV    *          PROBSAV
                               OUTPUT     FULL/SPEC
      (GENERATED)              BRANCH     *          MAJERR     MINERR
                               EXIT
      (GENERATED)              EXIT
      (GENERATED)   MAJERR     MESSAGE    MAJOR ERROR.  APEX TERMINATION
      (GENERATED)              EXIT
      (GENERATED)   MINERR     MESSAGE    MINOR ERROR.  APEX TERMINATION
      (GENERATED)              SAVE
      (GENERATED)              EXIT
      (GENERATED)   PROBOUT    MESSAGE    PROBLEM OUTPUT.  APEX TERMINATION
      (GENERATED)              BASISOU
      (GENERATED)              OUTPUT     FULL/SPEC
      (GENERATED)              SAVE
      (GENERATED)              EXIT
      (GENERATED)   PROBSAV    MESSAGE    PROBLEM SAVED.  APEX TERMINATION
      (GENERATED)              SAVE
      (GENERATED)              EXIT
      (GENERATED)   RECOUT     RECREAT
      (GENERATED)              BRANCH     PROBOUT    MAJERR
```

```
DATE 80/08/08  TIME 15.36.37          A P E X - I I I   E X E C U T I O N         APEX-III 1.202

           INPUT      $                                                                      ** APEX UNITS USED =      .459

                                             ***** CARD READ SUMMARY *****

      HEADER, CARD NO.       1    [NAME           ONTWAST                                                                  ]
      HEADER, CARD NO.       2    [ROWS                                                                                     ]
                                   FREE ROW = OPERATNG
                                   FREE ROW = TRANSPRT
                                   FREE ROW = REVENUES
                                   FREE ROW = NETCOST
                                   FREE ROW = CHCAPITL
                                   FREE ROW = CHOPRTNG
                                   FREE ROW = CHTRNPRT
                                   FREE ROW = CHREVENU
      HEADER, CARD NO.      81    [COLUMNS                                                                                  ]
      HEADER, CARD NO.     751    [RHS                                                                                      ]
                                   RHS NAME = ORIGRHS
                                   RHS NAME = CHGGEN
                                   RHS NAME = CHGDMD
                                   RHS NAME = CHGTLFL
                                   RHS NAME = CHGCPL
      HEADER, CARD NO.     779    [BOUNDS                                                                                   ]
                                   BOUND SET NAME = FACILITY
      HEADER, CARD NO.     792    [ENDATA                                                                                   ]
```

```
          **********   I N P U T   S U M M A R Y   S T A T I S T I C S   **********

      ***** GENERAL STATISTICS *****

  AUTO SELECTION       CONSTRAINTS             VARIABLES                NON-ZEROS                MISC. TOTALS
    NAME  = ONTWAST       EQ (E) ...      28     COLUMNS ...     120     AIJS (COL) ...     642    MINOR ERRORS     ...       0
    OBJ   = OPERATNG      LE (L) ...      30     RHS     ...       5     AIJS (RHS) ...      27    DENSITY 0/0      ...   6.859
    RHS   = ORIGRHS       GE (G) ...      12        TOTAL ...     125      TOTAL     ...     669    UNIQUE VALUES    ...     145
    BND   = FACILITY      FR (N) ...       8   MIN FL(8) ... 074100     AVER NZ/COL         5.35    INDIRECT NAMES   ...       0
    RNG   =                 TOTAL ...     78   REC FL(8) ... 105400     AVER NZ/ROW         8.23     TOTAL VALUES    ...     145
                         LC ROWS ...       0   ECS FL(8) ... 0021 K   LC AIJS (COL)..        0    PROBLEM BLOCKS   ...       1

      ***** BOUNDS STATISTICS *****

---- NAME = FACILITY    -----
FX      4  ZR      2  NZ      2
PL    116  LO      0  UP      8
MI      0  LO      0  UP      0
FR      0
BV      0

 I      0
S1      0  MB      0
S2      0  MB      0



      ***** RANGES STATISTICS *****

          NO SETS ACCEPTED

  NOTE- NROW NBND AND NRNG ARE RESIDENT IN CENTRAL MEMORY
        TOTAL LENGTH (LKVECNM) IS      92
```

```
           BRANCH     *          MAJERR     MINERR

           SET        KNDIR      MIN

           SET        KNOBJ      NETCOST

           SET        KNRHS      $

           SET        KNBND      $

           SET        KNRNG      $

           SELECT                                                                            ** APEX UNITS USED =     2.036


                                              ***** S E L E C T I O N *****

              ----------------- SELECTED VECTORS ------------------      ----- MULTIPLIERS ------   ----- DIRECTION ------
                     FUNCTION                CELL           NAME           CELL            VALUE

              OBJECTIVE FUNCTION         ( KNOBJ   )  =  NETCOST        ( RPSOBJ  )  =     1.000  ( KNDIR )  =  MINIMIZE
              CHANGE OBJECTIVE FUNCTION  ( KNCHOBJ )  =                 ( RPCHOBJ )  =     0.000

              RIGHT-HAND SIDE            ( KNRHS   )  =  ORIGRHS        ( RPSRHS  )  =     1.000
              CHANGE RIGHT-HAND SIDE     ( KNCHRHS )  =                 ( RPCHRHS )  =     0.000

              BOUNDS SET                 ( KNBND   )  =  FACILITY

              RANGES SET                 ( KNRNG   )  =

           BRANCH     *          MAJERR

           CRASH                                                                             ** APEX UNITS USED =     2.357

CRASH RESULTS - REMAINING LOGICALS =   56
                BALANCE STRUCTURALS =  16
                OTHER STRUCTURALS =     6

           BRANCH     *          MAJERR

           PRIMAL                                                                            ** APEX UNITS USED =     2.702

OPTIMAL SOLUTION.  OBJECTIVE =  -13250219.004595

           BRANCH     PRIMEND    MAJERR     *          PROBOUT

PRIMEND    BRANCH     *          MAJERR     PROBOUT

           MIXINT                                                                            ** APEX UNITS USED =     4.612

      PARENT                                                           - - - - - - - - SEPARATION VARIABLE -  - - - - - -
 NODE   NODE   ARR.                          OBJECTIVE       SUM OF    -                                               -              RESOURCE
  NO.    NO.   DIR.   STATUS            FUNCTION VALUE    FRACTIONS    - TYPE       NUMBER     LOWER LIMIT    UPPER LIMIT -   RANK       UNITS

  1       0           INTEGER          13250219.004595        0.000                                                           1          5.5
INTEGER SOLUTION FOUND AT NODE      1, WITH FUNCTIONAL VALUE     13250219.004595.
  IS ASSUMED OPTIMAL

           BRANCH     *          MAJERR     PROBSAV    *          PROBSAV
```

```
OUTPUT     FULL/SPEC                                        ** APEX UNITS USED =     6.057
```

```
DATE 80/08/08   TIME 15.36.39                    C O N S T R A I N T S                         APEX-III  1.202  PAGE      1

   PRINT OPTION = FULL        MAP NAME =                    SPECIAL/PDS = YES   VALUE OF OBJECTIVE =     -13250219.00460
NAME = ONTWAST     OBJ  = NETCOST     RHS  = ORIGRHS        BND = FACILITY    RPSOBJ   =      1.0000  RPSRHS   =      1.0000
DIR  = MINIMIZE    COBJ =             CRHS =                 RNG =             RPCHOBJ  =      0.0000  RPCHRHS  =      0.0000

    NUMBER     NAME     TYPE  STATUS     ROW ACTIVITY      SLACK        RHS LOWER      RHS UPPER       MARGINAL
    ------  ----------  ----  --------  -------------  -------------  -------------  -------------  -------------

         1  SRCGN121    EQ    BINDING    400000.00000        .         400000.00000   400000.00000        -.16374
         2  SRCGN125    EQ    BINDING    600000.00000        .         600000.00000   600000.00000        -.53567
         3  SRCGN126    EQ    BINDING    1700000.0000        .         1700000.0000   1700000.0000         .40657
         4  SRCGN141    EQ    BINDING    500000.00000        .         500000.00000   500000.00000        2.37829
         5  SRCGN152    EQ    BINDING    600000.00000        .         600000.00000   600000.00000        2.22580
         6  SRCGN154    EQ    BINDING    1060080.2000        .         1060080.2000   1060080.2000        2.20348
         7  TRFT0221    EQ    BINDING         .              .              .              .              -.53055
         8  TRLES221    LE     SLACK    -1819919.8000   1819919.8000      -INF             .                .
         9  TRMOS221    GE     SLACK     160080.20000  -160080.20000        .            +INF               .
        10  TRFRM221    EQ    BINDING         .              .              .              .              2.06958
        11  TRFT0233    EQ    BINDING         .              .              .              .             -1.09130
        12  TRLES233    LE     SLACK          .              .            -INF             .                .
        13  TRMOS233    GE     SLACK          .              .              .            +INF               .
        14  TRFRM233    EQ    BINDING         .              .              .              .              2.22208
        15  TRFT0243    EQ    BINDING         .              .              .              .              1.24525
        16  TRLES243    LE     SLACK          .              .            -INF             .                .
        17  TRMOS243    GE     SLACK          .              .              .            +INF               .
        18  TRFRM243    EC    BINDING         .              .              .              .              -.14018
        19  PLNT0321    EC    BINDING         .              .              .              .              -.18341
        20  PLEST321    LE     SLACK    -851479.39800   851479.39800      -INF             .                .
        21  PLMOS321    GE    BINDING         .              .              .            +INF            -4.53172
        22  PLFRM321    EQ    BINDING         .              .              .              .              -.59881
        23  PLAR3211    LE    BINDING         .              .            -INF             .             19.08035
        24  PLAR3212    LE     SLACK     -43004.01000    43004.01000      -INF             .                .
        25  PLAR3213    LE    BINDING         .              .            -INF             .             37.42056
        26  PLNT0333    EQ    BINDING         .              .              .              .               .05835
        27  PLEST333    LE     SLACK          .              .            -INF             .                .
        28  PLMOS333    GE    BINDING         .              .              .            +INF            -5.20866
        29  PLFRM333    EQ    BINDING         .              .              .              .              -.52442
        30  PLAR3331    LE    BINDING         .              .            -INF             .             18.96877
        31  PLAR3332    LE     SLACK          .              .            -INF             .                .
        32  PLAR3333    LE    BINDING         .              .            -INF             .             37.62512
        33  PLNT0343    EQ    BINDING         .              .              .              .             -2.38945
        34  PLEST343    LE     SLACK    -4000000.0000   4000000.0000      -INF             .                .
        35  PLMOS343    GE    BINDING         .              .              .            +INF            -3.24205
        36  PLFRM343    EQ    BINDING         .              .              .              .              -.66947
        37  PLAR3431    LE    BINDING         .              .            -INF             .             19.14358
        38  PLAR3432    LE     SLACK     -40000.00000    40000.00000      -INF             .                .
        39  PLAR3433    LE    BINDING         .              .            -INF             .             37.41312
        40  LFLT0419    EQ    BINDING         .              .              .              .               .55417
        41  LFLES419    LE    BINDING         .              .            -INF             .               .07553
        42  LFMOS419    GE    BINDING         .              .              .            +INF             -.39842
        43  LFLT0422    EQ    BINDING         .              .              .              .               .53558
        44  LFLES422    LE    BINDING         .              .            -INF             .               .27359
        45  LFMOS422    GE    BINDING         .              .              .            +INF             -.58601
        46  LFLT0432    EQ    BINDING         .              .              .              .               .46491
        47  LFLES432    LE     SLACK    -800000.00000   800000.00000      -INF             .                .
        48  LFMOS432    GE    BINDING         .              .              .            +INF             -.46491
        49  LFLT0434    EQ    BINDING         .              .              .              .               .49466
        50  LFLES434    LE     SLACK          .              .            -INF             .                .
```

```
DATE 80/08/08  TIME 15.36.39                   C O N S T R A I N T S                         APEX-III  1.202  PAGE     2

   PRINT OPTION = FULL        MAP NAME =                  SPECIAL/PDS = YES    VALUE OF OBJECTIVE =     -13250219.00460
NAME = ONTWAST      OBJ  = NETCOST      RHS  = ORIGRHS      BND = FACILITY     RPSOBJ  =      1.0000  RPSRHS  =       1.0000
DIR  = MINIMIZE     COBJ =              CRHS =              RNG =              RPCHOBJ =      0.0000  RPCHRHS =       0.0000

      NUMBER    NAME      TYPE   STATUS     ROW ACTIVITY      SLACK         RHS LOWER      RHS UPPER       MARGINAL
      ------ ----------   ----  --------   ------------- -------------   ------------- -------------   -------------

          51  LFMOS434     GE   BINDING          .              .              .            +INF            -.41802
          52  LFLT0441     EQ   BINDING          .              .              .              .              .57649
          53  LFLES441     LE   BINDING          .              .            -INF             .              .07339
          54  LFMOS441     GE    SLACK           .              .              .            +INF              .
          55  LFLT0444     EQ   BINDING          .              .              .              .              .63972
          56  LFLES444     LE    SLACK     -932838.49600   932838.49600      -INF             .               .
          57  LFMOS444     GE   BINDING          .              .              .            +INF            -.42818
          58  MAKT0156     EQ   BINDING          .              .              .              .          -18.57453
          59  MKMOS156     LE   BINDING     2000000.0000        .            -INF        2000000.0000        .02232
          60  MAKT0157     EQ   BINDING          .              .              .              .          -18.59684
          61  MKMOS157     LE    SLACK      430040.10000   1969959.9000      -INF        2400000.0000         .
          62  MAKT0258     EQ   BINDING          .              .              .              .              .66947
          63  MKMOS258     LE   BINDING     200000.00000        .            -INF        200000.00000      37.86316
          64  MAKT0339     EQ   BINDING          .              .              .              .          -37.19369
          65  MKMOS339     LE    SLACK       97201.60400   102798.39600      -INF        200000.00000         .
          66  LFLTOTAL     LE    SLACK      1932838.4960   998067160.50      -INF        999999999.00         .
          67  NUMTRANF     LE    SLACK           1.00000        4.00000      -INF             5.00000         .
          68  NUMPLANT     LE    SLACK           1.43004        3.56996      -INF             5.00000         .
          69  NUMLNDFL     LE    SLACK           1.46642        3.53358      -INF             5.00000         .
          70  CAPITALC     LE    SLACK      17575695.075   982424303.92      -INF        999999999.00         .
          71  OPERATNG     FR    SLACK      23204123.593  -23204123.593      -INF            +INF             .
          72  TRANSPRT     FR    SLACK      2215056.5758  -2215056.5758      -INF            +INF             .
          73  REVENUES     FR    SLACK     -56245100.089   56245100.089      -INF            +INF             .
          74  NETCOST      FR    SLACK     -13250219.005   13250219.005      -INF            +INF             .
          75  CHCAPITL     FR    SLACK      15790802.666  -15790802.666      -INF            +INF             .
          76  CHOPRTNG     FR    SLACK      18629778.017  -18629778.017      -INF            +INF             .
          77  CHTRNPRT     FR    SLACK      2215056.5758  -2215056.5758      -INF            +INF             .
          78  CHREVENU     FR    SLACK     -56245100.089   56245100.089      -INF            +INF             .
```

```
DATE 80/08/05  TIME 15.36.39                          C O L U M N S                              APEX-III  1.202  PAGE      1

   PRINT OPTION = FULL        MAP NAME =                   SPECIAL/POS = YES    VALUE OF OBJECTIVE =     -13250219.00460
NAME = ONTWAST      OBJ  = NETCOST      RHS  = ORIGRHS      BND = FACILITY     RPSOBJ  =       1.0000  RPSRHS  =      1.0000
DIR  = MINIMIZE     COBJ =              CRHS =              RNG =              RPCHOBJ =       0.0000  RPCHRHS =      0.0000

     NUMBER     NAME      TYPE   STATUS    COL ACTIVITY      OBJ COEF       BND LOWER      BND UPPER       MARGINAL
     ------  ----------   ----  --------  ------------    ------------    ------------   ------------   ------------

          1  ST121221      PL     ACTIVE   139919.80000          .67426          .              +INF              .
          2  ST125221      PL    LOWER          .             1.09102          .              +INF            .02480
          3  ST126221      PL     ACTIVE   1700000.0000          .12398          .              +INF              .
          4  ST126233      PL    LOWER          .             1.41336          .              +INF            .72863
          5  ST141221      PL    LOWER          .             1.73571          .              +INF           3.58345
          6  ST141233      PL    LOWER          .              .07439          .              +INF           1.36138
          7  ST141243      PL    LOWER          .             1.11581          .              +INF           4.73935
          8  ST152233      PL    LOWER          .             1.14061          .              +INF           2.27510
          9  ST152243      PL    LOWER          .              .12398          .              +INF           3.59502
         10  ST154233      PL    LOWER          .             1.31418          .              +INF           2.42636
         11  ST154243      PL    LOWER          .              .17357          .              +INF           3.62230
         12  SP121321      PL     ACTIVE   260080.20000          .34714          .              +INF              .
         13  SP121333      PL    LOWER          .             1.23979          .              +INF           1.13441
         14  SP125321      PL     ACTIVE   600000.00000          .71908          .              +INF              .
         15  SP125333      PL    LOWER          .             1.73571          .              +INF           1.25839
         16  SP126343      PL    LOWER          .             2.50438          .              +INF            .52150
         17  SP126321      PL    LOWER          .              .01488          .              +INF            .23804
         18  SP126333      PL    LOWER          .              .09298          .              +INF            .55791
         19  SP141321      PL    LOWER          .              .31243          .              +INF           2.50731
         20  SP141333      PL    LOWER          .             1.73571          .              +INF           4.17235
         21  SP141343      PL     ACTIVE   500000.00000          .01116          .              +INF              .
         22  SP152321      PL    LOWER          .              .16737          .              +INF           2.20976
         23  SP152333      PL    LOWER          .             1.05622          .              +INF           3.35037
         24  SP152343      PL     ACTIVE   600000.00000          .16365          .              +INF              .
         25  SP154321      PL    LOWER          .              .01860          .              +INF           2.03867
         26  SP154333      PL    LOWER          .             2.23162          .              +INF           4.49345
         27  SP154343      PL     ACTIVE   1060080.2000          .18597          .              +INF              .
         28  SL121419      PL    LOWER          .              .04463          .              +INF            .43507
         29  SL121422      PL    LOWER          .              .11158          .              +INF            .48342
         30  SL121432      PL    LOWER          .              .18597          .              +INF            .48714
         31  SL125419      PL    LOWER          .              .34714          .              +INF            .36564
         32  SL125422      PL    LOWER          .             1.04142          .              +INF           1.04133
         33  SL125432      PL    LOWER          .              .23804          .              +INF            .16728
         34  SL126419      PL    LOWER          .              .52071          .              +INF           1.48145
         35  SL126422      PL    LOWER          .              .08183          .              +INF           1.02397
         36  SL126432      PL    LOWER          .              .14878          .              +INF           1.02025
         37  SL141422      PL    LOWER          .             1.43816          .              +INF           4.35202
         38  SL141434      PL    LOWER          .              .17357          .              +INF           3.04652
         39  SL141444      PL    LOWER          .              .47112          .              +INF           3.48913
         40  SL152432      PL    LOWER          .              .09670          .              +INF           2.76741
         41  SL152444      PL    LOWER          .              .10414          .              +INF           2.96966
         42  SL152441      PL    LOWER          .              .11902          .              +INF           2.92131
         43  SL154432      PL    LOWER          .              .64306          .              +INF           3.51145
         44  SL154434      PL    LOWER          .              .24796          .              +INF           2.94610
         45  SL154441      PL    LOWER          .             1.53734          .              +INF           4.31731
         46  SL154444      PL    LOWER          .              .19837          .              +INF           3.04156
         47  TP221333      PL    LOWER          .              .16365          .              +INF           2.29159
         48  TP221343      PL     ACTIVE   1839919.8000          .31987          .              +INF              .
         49  TP233321      PL    LOWER          .              .14878          .              +INF           2.18745
         50  TP233343      PL     ACTIVE        .              .16737          .              +INF              .
```

```
DATE 80/08/06  TIME 15.36.39                    C O L U M N S                          APEX-III  1.202  PAGE      2

   PRINT OPTION = FULL         MAP NAME =                SPECIAL/PDS = YES     VALUE OF OBJECTIVE =     -13250219.00460
NAME = ONTWAST      OBJ  = NETCOST      RHS   = ORIGRHS     BND = FACILITY     RPSOBJ   =      1.0000  RPSRHS   =       1.0000
DIR  = MINIMIZE     COBJ =              CRHS  =             RNG =              RPCHOBJ  =      0.0000  RPCHRHS  =       0.0000

    NUMBER     NAME       TYPE  STATUS    COL ACTIVITY     OBJ COEF      BND LOWER     BND UPPER      MARGINAL
    ------  ----------    ----  --------  ------------   ------------  ------------  ------------  ------------

        51  TP243321      PL     ACTIVE        .               .32359        .             +INF           .
        52  TP243333      PL     ACTIVE        .               .08183        .             +INF           .
        53  TL221419      PL    LOWER          .               .04463        .             +INF        2.66839
        54  TL221422      PL    LOWER          .               .05951        .             +INF        2.66467
        55  TL221432      PL    LOWER          .               .13762        .             +INF        2.67211
        56  TL233419      PL    LOWER          .               .29755        .             +INF        3.07380
        57  TL233422      PL    LOWER          .               .20829        .             +INF        2.96594
        58  TL233432      PL    LOWER          .               .04463        .             +INF        2.73162
        59  TL233434      PL    LOWER          .               .07811        .             +INF        2.79485
        60  TL233441      PL    LOWER          .               .32359        .             +INF        3.12215
        61  TL233444      PL    LOWER          .               .13597        .             +INF        3.04776
        62  TL243422      PL    LOWER          .               .26780        .             +INF         .66319
        63  TL243432      PL    LOWER          .               .16365        .             +INF         .48838
        64  TL243434      PL    LOWER          .               .17853        .             +INF         .53301
        65  TL243441      PL    LOWER          .               .06323        .             +INF         .49954
        66  TL243444      PL    LOWER          .               .02976        .             +INF         .52929
        67  PL321419      PL     ACTIVE        .               .04463        .             +INF           .
        68  PL321422      PL     ACTIVE        .               .06323        .             +INF           .
        69  PL321432      PL     ACTIVE   412838.49600         .13390        .             +INF           .
        70  PL333419      PL    LOWER          .               .18597        .             +INF         .21572
        71  PL333422      PL    LOWER          .               .10414        .             +INF         .11530
        72  PL333432      PL     ACTIVE        .               .05951        .             +INF           .
        73  PL333434      PL    LOWER          .               .07439        .             +INF         .04463
        74  PL333441      PL    LOWER          .               .29755        .             +INF         .34962
        75  PL333444      PL    LOWER          .               .18597        .             +INF         .30127
        76  PL343422      PL    LOWER          .               .26780        .             +INF         .13390
        77  PL343432      PL     ACTIVE   587161.50400         .20457        .             +INF           .
        78  PL343434      PL     ACTIVE        .               .17481        .             +INF           .
        79  PL343441      PL     ACTIVE        .               .09298        .             +INF           .
        80  PL343444      PL     ACTIVE   932838.49600         .02976        .             +INF           .
        81  PM321156      PL     ACTIVE   430040.10000         .09298        .             +INF           .
        82  PM321157      PL    LOWER          .               .16737        .             +INF         .05207
        83  PM321258      PL    LOWER          .                .            .             +INF         .07067
        84  PM321339      PL     ACTIVE    17201.60400         .37194        .             +INF           .
        85  PM333156      PL     ACTIVE        .               .13018        .             +INF           .
        86  PM333157      PL    LOWER          .               .27395        .             +INF         .12646
        87  PM333258      PL    LOWER          .                .            .             +INF         .14506
        88  PM333339      PL     ACTIVE        .               .09298        .             +INF           .
        89  PM343156      PL     ACTIVE   1559959.9000         .10042        .             +INF           .
        90  PM343157      PL     ACTIVE   430040.10000         .12274        .             +INF           .
        91  PM343258      PL     ACTIVE   200000.00000          .            .             +INF           .
        92  PM343339      PL     ACTIVE    30000.00000         .45004        .             +INF           .
        93  WTPF221       PL     ACTIVE   1839919.8000        1.53904        .             +INF           .
        94  WTPF233       PL     ACTIVE        .              1.13076        .             +INF           .
        95  WTPF243       PL     ACTIVE        .              1.10507        .             +INF           .
        96  WPLT321       PL     ACTIVE   860069.20000        4.97465        .             +INF           .
        97  WPLT333       PL    LOWER          .              7.10813        .             +INF        2.54596
        98  WPLT343       PL     ACTIVE   4000000.0000        4.05256        .             +INF           .
        99  WLFL419       PL     ACTIVE        .               .23128        .             +INF           .
       100  WLFL422       PL     ACTIVE        .               .22316        .             +INF           .
```

```
DATE 30/08/98  TIME 15.35.39                       C O L U M N S                              APEX-III  1.202  PAGE      3

   PRINT OPTION = FULL         MAP NAME =                    SPECIAL/PDS = YES    VALUE OF OBJECTIVE =     -13250219.00460
NAME = ONTWAST       OBJ  = NETCOST       RHS   = ORIGRHS      BND = FACILITY    RPSOBJ   =      1.0000  RPSRHS   =       1.0000
DIR  = MINIMIZE      COBJ =               CRHS =               RNG =             RPCHOBJ  =      0.0000  RPCHRHS  =       0.0000

    NUMBER      NAME      TYPE  STATUS   COL ACTIVITY     OBJ COEF      BND LOWER      BND UPPER      MARGINAL
    ------  ----------  ----  -------- ------------  ------------  ------------  ------------  ------------

       101  WLFL432      PL    ACTIVE   1000000.0000         .              .          +INF             .
       102  WLFL434      PL    ACTIVE         .            .07664           .          +INF             .
       103  WLFL441      PL    ACTIVE         .            .64988           .          +INF             .
       104  WLFL444      PL    ACTIVE   932838.49600       .21154           .          +INF             .
       105  WMKT156      PL    ACTIVE   2000000.0000    -18.59684           .          +INF             .
       106  WMKT157      PL    ACTIVE   430040.10000    -18.59684           .          +INF             .
       107  WMKT258      PL    ACTIVE   200000.00000    -37.19369           .          +INF             .
       108  WMKT339      PL    ACTIVE    97201.60400    -37.19369           .          +INF             .
       109  DTRF221     BFX   LOWER          1.00000  1003731.4000       1.00000       1.00000  1003731.4000
       110  DTRF233      FX   LOWER           .       1362939.1000          .             .      1362939.1000
       111  DTRF243     BPL   LOWER           .       1732835.0000          .          1.00000  1732835.0000
       112  DPLT321     BPL    ACTIVE        .43004   9063445.7000          .          1.00000          .
       113  DPLT333     BPL    ACTIVE         .       5206657.4000          .          1.00000          .
       114  DPLT343     BPL     UPPER        1.00000  10034119.900          .          1.00000  -2934077.1900
       115  DLFL419     BPL    ACTIVE         .       796843.40000          .          1.00000          .
       116  DLFL422     BPL    ACTIVE         .       1054813.0000          .          1.00000          .
       117  DLFL432     BFX   LOWER          1.00000  1927052.9000       1.00000       1.00000  1462143.8500
       118  DLFL434      FX   LOWER           .       1370982.0000          .             .      952959.95000
       119  DLFL441     BPL   LOWER           .       108998.10000          .          1.00000  108998.10000
       120  DLFL444     BPL    ACTIVE        .46642   856360.10000          .          1.00000          .
```

```
          BRANCH     *          MAJERR     MINERR

          EXIT                                                                    ** APEX UNITS USED =     0.200

 --- USED WORDS OF DISK (OCTAL) - 000056243
```

APPENDIX 6

OUTPUT FROM RWP

```
      LITTLE  KINGSTON-QUINTE                   TEST DATA
   19812000   2
          2   6   3   3   6   3   2   1   1   0   0   0   0   0   0
     .03
     0
   PACKER  LONGHAL
            2.00          .30
   121 125 126 141 152 154
  SIDNEY  TRENTON BELLEVL NAPANEE KINGENV KINGSTON
     4860080.17
   221 233 243
  BELLEVILNAPANEE KINGSTON
       277517.99    263183.34    185577.65          .53          .35          .45    726213.43   1099755.79   1547257.39         1.01
             .78          .65
   321 333 343
  BELLVILENAPAN   KINGSTON
      3478345.32   5208657.40   3270967.60         2.45         0.00         2.23   5585100.39         0.00   6763152.33         2.53
            7.11         1.82
   419 422 432 434 441 444
  SIDNEY-ITYENDINGRICHMONDERNESTWNSTORRNT KINGSTON
       110000.00    459713.97    439305.45     76641.67    108998.08     95405.41          .05         0.00         0.00          .08
             .04          .05    686843.42    595098.99   1487747.49   1294340.31         0.00    760954.66          .18          .22
            0.00         0.00          .61          .16
   156 157
  PICTON  BATH
          -18.60       -18.60
   258
  HAMILTON
          -37.19
   339
  TORONTO
          -37.19
     11  27  46  52  66  80  92
   121 125 126 126 141 141 141 152 152 154 154 121 121 125 125 126 126 126 141 141
   141 152 152 152 154 154 154 121 121 121 125 125 125 126 126 126 141 141 141 152
   152 152 154 154 154 154 221 221 233 233 243 243 221 221 221 233 233 233 233 233
   233 243 243 243 243 243 321 321 321 333 333 333 333 333 333 343 343 343 343 343
   321 321 321 321 333 333 333 333 343 343 343 343
   221 221 221 233 221 233 243 233 243 233 243 321 333 321 333 343 321 333 321 333
   343 321 333 343 321 333 343 419 422 432 419 422 432 419 422 432 422 434 444 432
   444 441 432 434 441 444 333 343 321 343 321 333 419 422 432 419 422 432 434 441
   444 422 432 434 441 444 419 422 432 419 422 432 434 441 444 422 432 434 441 444
   156 157 258 339 156 157 258 339 156 157 258 339
             .69         1.09          .12         1.41         1.74          .07         1.12         1.14          .12         1.31
             .17          .35         1.24          .72         1.74         2.50          .01          .09          .31         1.74
             .01          .17         1.07          .16          .02         2.23          .19          .04          .11          .19
             .35         1.04          .24          .52          .08          .15         1.44          .17          .47          .10
             .10          .12          .84          .25         1.54          .20          .16          .32          .15          .17
             .32          .09          .04          .05          .14          .30          .21          .04          .08          .32
             .19          .27          .16          .18          .06          .03          .04          .06          .13          .19
             .10          .06          .07          .30          .19          .27          .20          .17          .09          .03
             .09          .17         0.00          .37          .13          .28         0.00          .09          .10          .12
            0.00          .45
NTWAST   NETCOST   ORIGRHS   FACILITY    P    1.000P    1.000M 13250219.004595280
                               P  0.000P    0.000       73       120
  CAPITALC        .175756950+08        .175756950+08
  OPERATNG        .232041240+08        .232041240+08
  TRANSPRT        .221505660+07        .221505660+07
  TOTCOSTS        .429948750+08        .429948750+08
  REVENUES       -.562451000+08       -.562451000+08
  NETCOST        -.132502190+08       -.132502190+08
```

```
ST121221          .13991980D+06
ST125221         0.
ST126221          .17000000D+07
ST126233         0.
ST141221         0.
ST141233         0.
ST141243         0.
ST152233         0.
ST152243         0.
ST154233         0.
ST154243         0.
SP121321          .26008020D+06
SP121333         0.
SP125321          .60000000D+06
SP125333         0.
SP126343         0.
SP126321         0.
SP126333         0.
SP141321         0.
SP141333         0.
SP141343          .50000000D+06
SP152321         0.
SP152333         0.
SP152343          .60000000D+06
SP154321         0.
SP154333         0.
SP154343          .10600802D+07
SL121419         0.
SL121422         0.
SL121432         0.
SL125419         0.
SL125422         0.
SL125432         0.
SL126419         0.
SL126422         0.
SL126432         0.
SL141422         0.
SL141434         0.
SL141444         0.
SL152432         0.
SL152444         0.
SL152441         0.
SL154432         0.
SL154434         0.
SL154441         0.
SL154444         0.
TP221333         0.
TP221343          .18399198D+07
TP233321         0.
TP233343         0.
TP243321         0.
TP243333         0.
TL221419         0.
TL221422         0.
TL221432         0.
TL233419         0.
TL233422         0.
TL233432         0.
TL233434         0.
TL233441         0.
TL233444         0.
TL243422         0.
TL243432         0.
TL243434         0.
TL243441         0.
TL243444         0.
```

```
PL321419        0.
PL321422        0.
PL321432         .41283850D+06
PL333419        0.
PL333422        0.
PL333432        0.
PL333434        0.
PL333441        0.
PL333444        0.
PL343422        0.
PL343432         .58716150D+06
PL343434        0.
PL343441        0.
PL343444         .93283850D+06
PM321156         .43004010D+06
PM321157        0.
PM321258        0.
PM321339         .17201604D+05
PM333156        0.
PM333157        0.
PM333258        0.
PM333339        0.
PM343156         .15699599D+07
PM343157         .43004010D+06
PM343258         .20000000D+06
PM343339         .80000000D+05
WTRF221          .18399198D+07
WTRF233         0.
WTRF243         0.
WPLT321          .86008020D+06
WPLT333         0.
WPLT343          .40000000D+07
WLFL419         0.
WLFL422         0.
WLFL432          .10000000D+07
WLFL434         0.
WLFL441         0.
WLFL444          .93283850D+06
WMKT156          .20000000D+07
WMKT157          .43004010D+06
WMKT258          .20000000D+06
WMKT339          .97201604D+05
DTRF221          .10000000D+01
DTRF233         0.
DTRF243         0.
DPLT321          .43004010D+00
DPLT333         0.
DPLT343          .10000000D+01
DLFL419         0.
DLFL422         0.
DLFL432          .10000000D+01
DLFL434         0.
DLFL441         0.
DLFL444          .46641925D+00
```

```
ONTARIO MINISTRY OF THE ENVIRONMENT, WASTE MANAGEMENT SYSTEMS MODEL          LITILE  KINGSTON-QUINTE                80/08/20.   PAGE   1

EXECUTIVE SUMMARY

IDENTIFICATION OF THE RUN:          TEST DATA

TIME PERIOD OF THE RUN:  1981 TO 2000

SYSTEM FINANCIAL SUMMARY                                                     MATERIAL FLOW IN THE SYSTEM

                          STUDY        ANNUAL        UNIT      PERCENT                                      STUDY        ANNUAL         PERCENT
                          PERIOD       AVERAGE                 OF TOTAL                                     PERIOD       AVERAGE        OF TOTAL
                           ($)         ($/YR)        ($/T)      (%)                                          ($)         ($/YR)          (%)

CAPITAL COST          17575695.08   878784.75     3.61634      34.0          WASTE GENERATED        4860080.17   243004.01       100.0
OPERATING COST        31193631.73  1559681.59     5.41834      60.3          WASTE RECOVERED        2727241.70   136362.09        56.1
TRANSPORT COST         2977731.90   148886.60      .61269       5.8          WASTE DISPOSED         2132838.47   106641.92        43.9
TOTAL COST            51747058.71  2587352.94    10.64737     100.0
REVENUE               75611084.03  3780554.20    15.55758
NET PROFIT            23864025.32  1193201.27     4.91021

 FACILITIES AND MARKETS USED IN THE SYSTEM

 TRANSFER STATIONS     USED          PROCESSING PLANTS     USED          LANDFILLS     USED           MARKETS      USED

     BELLEVIL           YES               BELLVILE          YES           SIDNEY-I      NO            PICTON        YES
     NAPANEE            NO                NAPAN             NO            TYENDING      NO            BATH          YES
     KINGSTON           NO                KINGSTON          YES           RICHMOND      YES           HAMILTON      YES
                                                                          ERNESTWN      NO            TORONTO       YES
                                                                          STORRNT       NO
                                                                          KINGSTON      YES

COSTS OF FACILITIES USED IN THE SYSTEM

                                          TRANSFER          PROCESSING          LANDFILL

                                            ($)                ($)                ($)

CAPITAL COST                              1254132.          15790805.           530761.
OPERATING COST                            3470078.          25044276.          2679280.
TRANSPORT COST                             791166.            666751.                0.
TOTAL COST OF FACILITIES                  5515375.          41501832.          3210041.
PERCENT OF SYSTEM TOTAL                      10.7               80.2               6.2
```

```
ONTARIO MINISTRY OF THE ENVIRONMENT, WASTE MANAGEMENT SYSTEMS MODEL          LITTLE  KINGSTON-QUINTE                 80/08/20.  PAGE  3

DETAILED REPORT OF WASTE GENERATION AREA SHIPMENTS

IDENTIFICATION OF THE RUN:        TEST DATA

TIME PERIOD OF THE RUN:  1981 TO 2000

GENERATION AREA(S) INCLUDED IN THIS REPORT:  ALL

  GENERATION       TRANSPORT        AVERAGE         UNIT         QUANTITY       AVERAGE
   AREA NAME          COST         TRANSPORT      TRANSPORT      OF WASTE        WASTE
                                    COST/YR.      COST          GENERATED      GEN./YR.
                       ($)            ($)         ($)                 (T)           (T)

   SIDNEY           251963.         12598.        .62991        400000.        20000.
   TRENTON          580000.         29000.        .96667        600000.        30000.
   BELLEVL          283333.         14167.        .16667       1700000.        85000.
   NAPANEE            7500.           375.        .01500        500000.        25000.
   KINGENV          132000.          6600.        .22000        600000.        30000.
   KINGSTON         265020.         13251.        .25000       1060080.        53004.

  SUBTOTAL TO      TRANSPORT        AVERAGE         UNIT         QUANTITY       AVERAGE
 SOURCE ID NO.        COST         TRANSPORT      TRANSPORT      OF WASTE        WASTE
                                    COST/YR.      COST          GENERATED      GEN./YR.
                       ($)            ($)         ($)                 (T)           (T)

      126          1115296.         55765.        .41307       2700000.       135000.
      152           139500.          6975.        .12682       1100000.        55000.
```

```
DETAILED REPORT OF FACILITY OPERATIONS

IDENTIFICATION OF THE RUN:         TEST DATA

TIME PERIOD OF THE RUN:  1981 TO 2000

FACILITIES INCLUDED IN THIS REPORT:  ALL

                  ------------------ ANNUAL COSTS ------------------    ---------------- UNIT COSTS --------------

  FACILITY      CAPITAL       OPERATING      TRANSPORT     TOTAL              CAPITAL       OPERATING       TRANSPORT    TOTAL          WASTE
                  COST            COST           COST      COST                COST            COST            COST     COST         SHIPPED
                  ($)             ($)            ($)       ($)                ($/T)           ($/T)           ($/T)    ($/T)           (T)


   BELLEVIL      1254132.       3470078.       791166.     5515375.           .68162         1.88599         .43000    2.99762       1839920.
   NAPANEE            0.             0.            0.           0.          0.00000         0.00000        0.00000    0.00000             0.
   KINGSTON           0.             0.            0.           0.          0.00000         0.00000        0.00000    0.00000             0.
   BELLVILE      3600611.       6151089.       136667.     9888367.          4.18637         7.15176         .15890   11.49703        860080.
   NAPAN              0.             0.            0.           0.          0.00000         0.00000        0.00000    0.00000             0.
   KINGSTON     12190193.      18893187.       530084.    31613465.          3.04755         4.72330         .13252    7.90337       4000000.
   SIDNEY-I           0.             0.            0.           0.          0.00000         0.00000        0.00000    0.00000             0.
   TYENDING           0.             0.            0.           0.          0.00000         0.00000        0.00000    0.00000             0.
   RICHMOND       439305.       2000000.            0.     2439305.           .43931         2.00000        0.00000    2.43931       1000000.
   ERNESTWN           0.             0.            0.           0.          0.00000         0.00000        0.00000    0.00000             0.
   STORRNT            0.             0.            0.           0.          0.00000         0.00000        0.00000    0.00000             0.
   KINGSTON        91456.        679280.            0.      770736.           .09804          .72819        0.00000     .82623        932838.
```

```
ONTARIO MINISTRY OF THE ENVIRONMENT, WASTE MANAGEMENT SYSTEMS MODEL       LITTLE  KINGSTON-QUINTE                 80/08/20.  PAGE  5

MARKETING REPORT

IDENTIFICATION OF THE RUN:        TEST DATA

TIME PERIOD OF THE RUN:  1981 TO 2000
   MARKET NAME     PRODUCT NAME     QUANTITY           MARKET PRICE       REVENUE
                                       (T)                  ($)              ($)

     PICTON          R.D.F.         2000000.                25.          50000000.
     BATH            R.D.F.          430040.                25.          10751003.
     HAMILTON        FERROUS         200000.                50.          10000000.
     TORONTO         GLASS           430040.                50.          21502005.

                     TOTALS         3060080.                             92253007.
```

```
DETAILED REPORT ON TRANSFER STATIONS AND PLANTS SHIPMENTS

IDENTIFICATION OF THE RUN:         TEST DATA

TIME PERIOD OF THE RUN:  1981 TO 2000

         FACILITY TYPES:  PL -- PROCESSING PLANT,  LF -- LANDFILL,   MK -- MARKET,  TR -- TRANSFER STN.


    QUANTITIES ARE IN THOUSANDS OF TONNES.  TOTAL COST AND YEARLY AVERAGE COST ARE
    IN THOUSANDS OF DOLLARS, AVERAGE IN DOLLARS PER TONNE.


   ---- FROM ----    ----------------------------    SHIPPED TO FACILITY   ----------------------------
   FACILITY  TYPE      NAME      TYPE        TOTAL          YEARLY          TOTAL            ANNUAL         AVERAGE
                                           QUANTITY        AVERAGE          COST             AVERAGE        COST/TON

   BELLEVIL   TR    KINGSTON     PL         1839.9          92.0            791.2              39.6          .43000
   BELLVILE   PL    RICHMOND     LF          412.5          20.6             74.3               3.7          .18000
   BELLVILE   PL    PICTON       MK          430.0          21.5             53.8               2.7          .12500
   BELLVILE   PL    TORONTO      MK           17.2            .9              8.6                .4          .50000
   KINGSTON   PL    RICHMOND     LF          587.2          29.4            161.5               8.1          .27500
   KINGSTON   PL    KINGSTON     LF          932.8          46.6             37.3               1.9          .04000
   KINGSTON   PL    PICTON       MK         1570.0          78.5            211.9              10.6          .13500
   KINGSTON   PL    BATH         MK          430.0          21.5             71.0               3.5          .16500
   KINGSTON   PL    HAMILTON     MK          200.0          10.0              0.0               0.0         0.00000
   KINGSTON   PL    TORONTO      MK           80.0           4.0             48.4               2.4          .60500
```

APPENDIX 7

SOURCE CODE OF ESP FOR CDC

```
      PROGRAM OWASTE(TAPE1,TAPE5,TAPE6,TAPE9)
C   MODIFICATION FOR CDC
C AUGUST 8, 1980
C      *****************************************************************
C      *                                                               *
C      *       THE  ONTARIO  WASTE  MANAGEMENT  SYSTEMS  MODEL         *
C      *                                                               *
C      *****************************************************************
C      *                                                               *
C      *  THE MODEL CONSISTS OF TWO FORTRAN PROGRAMS TO BE USED IN     *
C      *  CONJUNCTION WITH A MIXED INTEGER LINEAR PROGRAMMING PACKAGE  *
C      *  SUCH AS APEX-III OF CDC OR MPSX-MIP OF IBM.                  *
C      *                                                               *
C      *  THIS PROGRAM, THE EDIT AND SETUP PROGRAM, READS IN RAW DATA  *
C      *  AND CALCULATES VALUES TO ENTER THE MILP. THE PROGRAM PRINTS  *
C      *  OUT RAW AND CALCULATED DATA AND ALSO CHECKS FOR OBVIOUS      *
C      *  DATA ERRORS.                                                 *
C      *  IF NO ERRORS ARE DETECTED, THE PROGRAM MAY WRITE FILE        *
C      *  ONTWAST FOR ENTRY TO THE MILP PACKAGE, AND ALSO A FILE       *
C      *  TO BE USED, IN CONJUNCTION WITH OUTPUT FROM THE MILP, BY     *
C      *  THE REPORT WRITER PROGRAM.                                   *
C      *                                                               *
C      *            PROGRAMMED BY L. JENKINS.     DECEMBER, 1979.      *
C      *            MODIFIED BY J. POLAK          JULY, 1980           *
C      *                                                               *
C      *                                                               *
C      *                                                               *
C      *                                                               *
C      *   UNIT 1 - DATA FOR APEX - T FILE                             *
C      *   UNIT 9 - DATA FOR REPORT GENERATOR - D FILE                 *
C      *   UNIT 5 - DATA INPUT                                         *
C      *   UNIT 6 - PRINTED OUTPUT                                     *
C      *                                                               *
C      *                                                               *
C      *****************************************************************
C
C      ***   INITIAL DECLARATION OF L VARIABLES AS REAL
C
       IMPLICIT REAL(L)
C
C      ***   DECLARATION OF VARIABLES FOR TITLES
C             AND NAMES
       REAL ASTRSK
       REAL TITLE,RUNNAM,VECNAM,SRCNAM,TRFNAM,PLTNAM,
     1       LFLNAM,RSCNAM,MKTNAM
C
       INTEGER R,W,T,D
       INTEGER TOTERR,ERFLAG,RUNYRS,
     1          RUNBEG,RUNEND,INREP,OUTREP,TWRITE,DWRITE,LANDCD,
     2          SRCION,TRFION,PLTION,LFLION,MKTION,
     3           ORIGID,DESTID,TRSTAT,PLSTAT,LFSTAT,STATJ
C
C      *** GENERAL COMMON BLOCK
C
       COMMON/GENDAT/ASTRSK(15),TITLE(5),RUNNAM(5),
     1               LTOLFL,LNUMTR,LNUMPL,LNUMLF,LCAPTL,RATE,
     2               R,W,T,D,TOTERR,ERFLAG,DAT,
     3               RUNBEG,RUNEND,RUNYRS,INREP,OUTREP,TWRITE,DWRITE,
     4               LANDCD,NUMVEC,NUMSRC,NUMTRF,NUMPLT,NUMLFL,NRESRC,
     5               NUMMKT(9),NPAGE,LINPAG
C
C
C      ***   INITIALIZE DEVICE NUMBERS
C
      R=5
      W=6
      T=1
      D=9
      LINPAG=54
      NPAGE=0
C
C
         CALL DATE(DAT)
         CALL START
         CALL SOURCE
         IF(NUMTRF.GT.0) CALL TRANFR
         IF(NUMPLT.GT.0) CALL PLANTS
         CALL LANDFL
         IF(NRESRC.GT.0) CALL MARKET
         CALL LINKEM
         IF(TWRITE.EQ.1) CALL TSETUP
         IF(DWRITE.EQ.1) CALL DSETUP
         STOP
         END
      SUBROUTINE HEADER(NROUTN)
C
C        ***********************************************************
C        *                                                         *
C        *  THIS ROUTINE WRITES A HEADING AT THE BEGINNING OF OUTPUT *
C        *    FROM EACH ROUTINE                                    *
C        *                                                         *
C        ***********************************************************
C
C     ***   INITIAL DECLARATION OF L VARIABLES AS REAL
C
      IMPLICIT REAL(L)
C
C     ***   DECLARATION OF VARIABLES FOR NAMES AND
C           TITLES
      REAL ASTRSK
      REAL TITLE,RUNNAM,VECNAM,SRCNAM,TRFNAM,PLTNAM,
     1      LFLNAM,RSCNAM,MKTNAM
C
      INTEGER R,W,T,D
      INTEGER TOTERR,ERFLAG,RUNYRS,
     1          RUNBEG,RUNEND,INREP,OUTREP,TWRITE,DWRITE,LANDCD,
     2           SRCIDN,TRFIDN,PLTIDN,LFLIDN,MKTIDN,
     3          ORIGID,DESTID,TRSTAT,PLSTAT,LFSTAT,STATJ
      REAL RTNNAM(9)
C
C     *** GENERAL COMMON BLOCK
C
      COMMON/GENDAT/ASTRSK(15),TITLE(5),RUNNAM(5),
     1             LTOLFL,LNUMTR,LNUMPL,LNUMLF,LCAPTL,RATE,
     2             R,W,T,D,TOTERR,ERFLAG,DAT,
     3             RUNBEG,RUNEND,RUNYRS,INREP,OUTREP,TWRITE,DWRITE,
     4             LANDCD,NUMVEC,NUMSRC,NUMTRF,NUMPLT,NUMLFL,NRESRC,
     5             NUMMKT(9),NPAGE,LINPAG
C
      DATA RTNNAM/"START","SOURCE","TRANFR","PLANTS","LANDFL",
     1            "MARKET","LINKEM","TSETUP","DSETUP"/
C
      NPAGE=NPAGE+1
      WRITE(W,101) ASTRSK,NPAGE,DAT
 101  FORMAT("1",15A8,1X,"PAGE NO.",I3/" *",118X,"*",A10)
      WRITE(W,102)
 102  FORMAT(" *",5X,"ONTARIO MINISTRY OF THE ENVIRONMENT --",
     1      " WASTE MANAGEMENT SYSTEMS MODEL-VERSION 1.1 --",
     2 " EDIT AND SETUP PROGRAM",6X,"*"/" *",118X,"*")
C
      WRITE(W,111) TITLE,RUNNAM,RTNNAM(NROUTN),ASTRSK
 111  FORMAT(" *",5X,5A8,5A8,"  OUTPUT FROM ROUTINE ",A6,5X,"*"/
     1      " *",118X,"*"/" ",15A8/)
C
      RETURN
      END
      SUBROUTINE ERROR(NEROR)
C
C      *****************************************************************
C      *                                                               *
C      *    THIS ROUTINE PRINTS THE ERROR NUMBER AND SETS FLAGS        *
C      *    TWRITE AND DWRITE TO 0.                                    *
C      *    AFTER 10 ERRORS THE PROGRAM STOPS                          *
C      *                                                               *
C      *****************************************************************
C
C      ***    INITIAL DECLARATION OF L VARIABLES AS REAL
C
       IMPLICIT REAL(L)
C
C      ***    DECLARATION OF VARIABLES FOR NAMES AND
C             TITLES
       REAL ASTRSK
       REAL TITLE,RUNNAM,VECNAM,SRCNAM,TRFNAM,PLTNAM,
     1         LFLNAM,RSCNAM,MKTNAM
C
       INTEGER R,W,T,D
       INTEGER TOTERR,ERFLAG,RUNYRS,
     1          RUNBEG,RUNEND,INREP,OUTREP,TWRITE,DWRITE,LANDCD,
     2           SRCION,TRFION,PLTION,LFLION,MKTION,
     3          ORIGID,DESTID,TRSTAT,PLSTAT,LFSTAT,STATJ
C
C      *** GENERAL COMMON BLOCK
C
       COMMON/GENDAT/ASTRSK(15),TITLE(5),RUNNAM(5),
     1              LTOLFL,LNUMTR,LNUMPL,LNUMLF,LCAPTL,RATE,
     2              R,W,T,D,TOTERR,ERFLAG,DAT,
     3              RUNBEG,RUNEND,RUNYRS,INREP,OUTREP,TWRITE,DWRITE,
     4              LANDCD,NUMVEC,NUMSRC,NUMTRF,NUMPLT,NUMLFL,NRESRC,
     5              NUMMKT(9),NPAGE,LINPAG
C
C
C     AN ERROR RETURN FLAG MAY BE USED TO BYPASS THE ERRONEOUS CARD
      ERFLAG=1
      TWRITE=0
      DWRITE=0
      TOTERR=TOTERR+1
      IF(TOTERR.GT.10) GO TO 9999
      WRITE(W,1001)(ASTRSK(I),I=1,10),NEROR
 1001 FORMAT(/10A8,"   ERROR NUMBER ",I3/)
      RETURN
C
 9999 WRITE(W,9009)ASTRSK,ASTRSK
 9009 FORMAT(/15A8/40X," MORE THAN TEN ERRORS - I QUIT "/15A8)
      STOP
      END
      SUBROUTINE START
C
C        **********************************************************
C        *                                                        *
C        *  THIS ROUTINE READS IN ALL HEADER AND GENERAL          *
C        *       PARAMETER CARDS                                  *
C        *                                                        *
C        **********************************************************
C
C     ***    INITIAL DECLARATION OF L VARIABLES AS REAL
C
      IMPLICIT REAL(L)
C
C     ***    DECLARATION OF VARIABLES FOR NAMES AND
C            TITLES
      REAL ASTRSK,AST
      REAL TITLE,RUNNAM,VECNAM,SRCNAM,TRFNAM,PLTNAM,
     1       LFLNAM,RSCNAM,MKTNAM
C
      INTEGER R,W,T,D,LINES
      INTEGER TOTERR,ERFLAG,RUNYRS,
     1          RUNBEG, RUNEND,INREP,OUTREP,TWRITE,DWRITE,LANDCD,
     2           SRCIDN,TRFIDN,PLTIDN,LFLIDN,MKTIDN,
     3          ORIGID,DESTID,TRSTAT,PLSTAT,LFSTAT,STATJ
C
C     *** GENERAL COMMON BLOCK
C
      COMMON/GENDAT/ASTRSK(15),TITLE(5),RUNNAM(5),
     1              LTOLFL,LNUMTR,LNUMPL,LNUMLF,LCAPTL,RATE,
     2              R,W,T,D,TOTERR,ERFLAG,DAT,
     3              RUNBEG,RUNEND,RUNYRS,INREP,OUTREP,TWRITE,DWRITE,
     4              LANDCD,NUMVEC,NUMSRC,NUMTRF,NUMPLT,NUMLFL,NRESRC,
     5              NUMMKT(9),NPAGE,LINPAG
C
C     ***    VEHICLES COMMON BLOCK
C
      COMMON/VECDAT/VECNAM(8),COSTTH(8)
C
      REAL TINREP(12),TOUTRP(9),FIT(4),FID(4),NLFCS(10)
      REAL NOYES(2)
      DATA TINREP/"ERROR ME","SSAGES O","NLY     ","        ",
     1   "TRANSPOR","TATION L","INKS NOT"," LISTED ",
     2   "COMPLETE","        ","        ","        "/
      DATA TOUTRP/"EXECUTIV","E SUMMAR","Y ONLY  ","COMPLETE",
     1   "        ","        ","COMPLETE","        ","        "/
      DATA FIT /"        ","        ","WRITE FI","LE T    "/
      DATA FID /"        ","        ","WRITE FI","LE D    "/
      DATA NLFCS /"FIXED AN","D VARIAB","LE LANDF","ILL COST",
     1       "S       ","LINEAR A","PPROXIMA","TION OF ","LANDFILL",
     2       " COSTS  "/
      DATA NOYES/"NO  ","YES "/
      DATA AST/"********"/
C
C     *** CREATE LINE OF ASTERISKS
      DO 10 J=1,15
      ASTRSK(J)=AST
 10   CONTINUE
```

```
C
C       ***   INITIALIZE COUNTERS AND FLAGS TO 0
C
      TOTERR=0
      ERFLAG=0
      LINES=26
C
C
C       ***   READ TITLE,RUN DESCRIPTION
C
 1000 FORMAT(5A8)
      READ (R,1000)TITLE
      READ (R,1000)RUNNAM
      CALL HEADER(1)
C
      READ(R,1010)RUNBEG,RUNEND,RATE,INREP,OUTREP,TWRITE,DWRITE,LANDCD
 1010 FORMAT(2I5,F10.2,5(4X,I1))
      RUNYRS=RUNEND-RUNBEG+1
      IF(INREP.GT.0)WRITE(W,1014)RUNBEG,RUNEND,RUNYRS,RATE
 1014 FORMAT(10X,"STUDY PERIOD ",I4,"-",I4,"(",I2," YEARS)"
     1      /10X, "DISCOUNT RATE IS ",F5.2," % ")
      RATE=RATE/100.
 1020 FORMAT(10I5)
      READ(R,1020)NUMVEC,NUMSRC,NUMTRF,NUMPLT,NUMLFL,NRESRC
      READ(R,1020)NUMMKT
      READ (R,1100) LTOLFL,LCAPTL
 1100 FORMAT(5F10.0)
      IF(LTOLFL.EQ.0) LTOLFL=99999999
      IF(LCAPTL.EQ.0) LCAPTL=99999999
      READ(R,1100) LNUMTR,LNUMPL,LNUMLF
      NNLF=LNUMLF
      NNTR=LNUMTR
      NNPL=LNUMPL
      WRITE (W,1015) NUMSRC,(TINREP(INREP*4+I),I=1,4),NUMTRF,NNTR,
     1 (TOUTRP(OUTREP*3+I),I=1,3),NUMPLT,NNPL,(NLFCS(LANDCD*5+I),
     2 I=1,5),NUMLFL,NNLF,(FIT(TWRITE*2+I),I=1,2),
     3 (FID(DWRITE*2+I),I=1,2)
 1015 FORMAT (///T29,"GIVEN OR",T42,"MAXIMUM"/T29,"POSSIBLE",T42,
     1 "ALLOWED"/T29,"LOCATIONS",T42,"IN RUN",T70,"OPTIONS SELECTED",
     2 " IN THE RUN"//T10,"SOURCES",T29,I5,T70,"INPUT REPORT  - ",
     3 4A8/T10,"TRANSFER STATIONS",T29,I5,T40,I5,T70,
     4 "OUTPUT REPORT - ",3A8/T10,"PLANTS",T29,I5,T40,I5,T70,
     5 5A8/T10,"LANDFILLS",T29,I5,T40,I5,T70,2A8/T70,2A8)
      IF (INREP.GT.0) WRITE (W,1030) LTOLFL,LCAPTL
 1030 FORMAT (//9X,"LIMITS OVER THE PERIOD FOR:"/13X,
     1 "TOTAL LANDFILLING",4X,F12.0," TONNES"/13X,
     2 "CAPITAL EXPENDITURES $",F11.0)
      LINES=LINES+NRESRC
      IF (LINES.GT.LINPAG.AND.INREP.GT.0) CALL HEADER(1)
      IF (INREP.LE.0.OR.NRESRC.EQ.0) GO TO 110
      WRITE (W,1040)
 1040 FORMAT (//10X,"RESOURCE",5X,"NUMBER OF"/23X,"MARKETS"/)
      DO 20 J=1,NRESRC
   20 WRITE (W,1105) J,NUMMKT(J)
 1105 FORMAT (10X,I5,7X,I5)
  110 CONTINUE
C
```

```
C               READ IN VEHICLE DATA
C
      READ(R,1120)(VECNAM(I),COSTTH(I),I=1,NUMVEC)
 1120 FORMAT(4(2X,A8,F5.3))
      IF(INREP.EQ.0) GO TO 120
      LINES=LINES+7+NUMVEC
      IF (LINES.GT.LINPAG) CALL HEADER(1)
      WRITE (W,1130)
 1130 FORMAT (//10X,"VEHICLE DATA"//13X,"VEHICLE",7X,"NAME",
     1 12X,"COST PER TONNE"/13X,"NUMBER",24X,"PER HOUR, $"/)
      DO 117 I=1.NUMVEC
  117 WRITE (W,1140) I,VECNAM(I),COSTTH(I)
 1140 FORMAT (12X,I5.6X,A8,13X,F6.3)
C
 120  RETURN
      END
      SUBROUTINE SOURCE
C
C        ************************************************************
C        *                                                          *
C        *   THIS ROUTINE READS IN GENERATION DATA AND CALCULATES   *
C        *   TOTAL WASTE GENERATED AT EACH SOURCE OVER THE PERIOD   *
C        *                                                          *
C        ************************************************************
C
C     ***   INITIAL DECLARATION OF L VARIABLES AS REAL
C
      IMPLICIT REAL(L)
C
C     ***   DECLARATION OF VARIABLES FOR NAMES AND
C           TITLES
      REAL ASTRSK
      REAL TITLE,RUNNAM,VECNAM,SRCNAM,TRFNAM,PLTNAM,
     1         LFLNAM,RSCNAM,MKTNAM
C
      INTEGER R,W,T,D,LINES
      INTEGER TOTERR,ERFLAG,RUNYRS,
     1           RUNBEG,RUNEND,INREP,OUTREP,TWRITE,DWRITE,LANDCO,
     2            SRCIDN,TRFIDN,PLTIDN,LFLIDN,MKTIDN,
     3           ORIGID,DESTID,TRSTAT,PLSTAT,LFSTAT,STATJ
C
C     *** GENERAL COMMON BLOCK
C
      COMMON/GENDAT/ASTRSK(15),TITLE(5),RUNNAM(5),
     1               LTOLFL,LNUMTR,LNUMPL,LNUMLF,LCAPTL,RATE,
     2               R,W,T,D,TOTERR,ERFLAG,DAT,
     3               RUNBEG,RUNEND,RUNYRS,INREP,OUTREP,TWRITE,DWRITE,
     4               LANDCO,NUMVEC,NUMSRC,NUMTRF,NUMPLT,NUMLFL,NRESRC,
     5               NUMMKT(9),NPAGE,LINPAG
C
C     ***   SOURCES COMMON BLOCK
C
      COMMON/SRCDAT/SRCNAM(200),SRCTOT(200),ALWAST,SRCIDN(200)
C
C
C
      DIMENSION IYR(4),AMOUNT(4),TONYR(4),POPYR(4),NT(5)
      DIMENSION ITYPE(200),IYR3(4,200),AM3(4,200),NEN3(200)
      DIMENSION TONY3(4,200),POPY3(4,200),AMONT1(200),GROWTH(200)
      DIMENSION TONYR1(200),TONGRO(200),POPYR1(200),POPGRO(200)
C
     IF(INREP.GT.0) CALL HEADER(2)
C
      ALWAST=0
      LINES=10
      DO 5 K=1,5
    5 NT(K)=0.0
C
      DO 200 J=1,NUMSRC
      READ(R,2010)SRCIDN(J),SRCNAM(J),IGENTP
C
C  READ THE DATA IN AND STORE, FOR ALL TYPES
C
```

```
      ITYPE(J)=IGENTP
 2010 FORMAT(I3,A8,3X,I1)
      IF (IGENTP.LT.0.OR.IGENTP.GT.5) GO TO 101
C
      IF(IGENTP.NE.1) GO TO 120
C
C   TYPE 1
C
      NT(1)=NT(1)+1
      READ(R,2020)IDNUMB,TAMONT
 2020 FORMAT(I3,7X,F10.0)
      IF(IDNUMB.NE.SRCIDN(J)) GO TO 102
      SRCTOT(J)= TAMONT
      ALWAST = ALWAST + TAMONT
      GO TO 200
C
 120  IF(IGENTP.NE.2) GO TO 130
C
C   TYPE 2
C
      NT(2)=NT(2)+1
      READ(R,2040)IDNUMB,AMONT1(J),GROWTH(J)
 2040 FORMAT(I3,7X,F10.0,F10.2)
      IF(IDNUMB.NE.SRCIDN(J)) GO TO 102
C
C       ***  SUM GENERATION OVER WHOLE PERIOD
C
      GROW = 1+GROWTH(J)/100.
      NYRS = RUNYRS-1
      YRTOT = AMONT1(J)
      SRCTOT(J) = YRTOT
      DO 10 I=1,NYRS
      YRTOT = YRTOT*GROW
 10   SRCTOT(J) = SRCTOT(J)+YRTOT
C
      ALWAST =ALWAST +SRCTOT(J)
      GO TO 200
 130  IF(IGENTP.NE.3) GO TO 140
C
C   TYPE 3
C
      NT(3)=NT(3)+1
      READ (R,2060) IDNUMB,NEN3(J),(IYR3(I,J),AM3(I,J),I=1,4)
 2060 FORMAT(I3,6X,I1,4(I5,F10.0))
      IF(IDNUMB.NE.SRCIDN(J)) GO TO 102
      IF(IYR3(1,J).NE.RUNBEG) GO TO 103
      NE=NEN3(J)
      IF(IYR3(NE,J).NE.RUNEND) GO TO 104
      GO TO 200
 140  IF(IGENTP.NE.4) GO TO 150
C
C   TYPE 4
C
      NT(4)=NT(4)+1
      READ(R,2080)IDNUMB,TONYR1(J),TONGRO(J),POPYR1(J),POPGRO(J)
 2080 FORMAT(I3,7X,F5.3,F5.2,F10.0,F5.2)
      IF(IDNUMB.NE.SRCIDN(J)) GO TO 102
```

```
C
C       ***  SUM GENERATION OVER WHOLE PERIOD
C
        GROW =(1.+TONGRO(J)/100.) *(1. +POPGRO(J)/100)
        NYRS = RUNYRS-1
        YRTOT = TONYR1(J) * POPYR1(J)
        SRCTOT(J)=YRTOT
        DO 20 I=1,NYRS
        YRTOT = YRTOT * GROW
 20     SRCTOT(J) = SRCTOT(J) + YRTOT
C
        ALWAST =  ALWAST +SRCTOT(J)
        GO TO 200
C
C       ***   LAST GENERATION TYPE CONSIDERED - IE. TYPE 5
C
  150   NT(5)=NT(5)+1
        READ(R,2091)IDNUMB,NEN3(J),(IYR3(I,J),TONY3(I,J),
      1 POPY3(I,J),I=1,4)
 2091   FORMAT(I3,6X,I1,4(I5,F5.2,F7.0))
        IF(IDNUMB.NE.SRCIDN(J)) GO TO 102
        IF(IYR3(1,J).NE.RUNBEG) GO TO 103
        NE=NEN3(J)
        IF(IYR3(NE,J).NE.RUNEND) GO TO 104
        GO TO 200
C
C      ***   ERROR CALLS
 101  CALL ERROR(201)
      GO TO 200
 102  CALL ERROR(212)
      GO TO 200
 103  CALL ERROR(223)
      GO TO 200
 104  CALL ERROR(224)
 200  CONTINUE
C
C  WRITE THE RESULTS FOR ALL TYPES
C
C
C  PRINT THE RESULTS FORT TYPE 1 IF THERE ARE ANY
C
        IF (NT(1).EQ.0.OR.INREP.EQ.0) GO TO 230
        I1=1
        WRITE (W,2110) I1
 2110   FORMAT (/1X,"SOURCES - DATA TYPE",I2)
        WRITE (W,2029)
 2115   FORMAT (/1X,"SOURCES - DATA TYPE",I2," (CONTINUED)")
 2029   FORMAT (/T23,"TOTAL",T121,"TOTAL"/T11,"SOURCE",T23,"WASTE",
      1 T121,"WASTE"/T3,"LABEL",T12,"NAME",T23,"TONNES",T121,"TONNES"/)
        DO 210 J=1,NUMSRC
        IF (ITYPE(J).NE.1) GO TO 210
        IF (LINES.LE.LINPAG) GO TO 220
        CALL HEADER(2)
        WRITE (W,2115) I1
        WRITE (W,2029)
        LINES=10
  220   WRITE (W,2030) SRCIDN(J),SRCNAM(J),SRCTOT(J),SRCTOT(J)
 2030 FORMAT (1X,I5,3X,A8,2X,F10.0,T117,F10.0)
      LINES=LINES+1
  210 CONTINUE
      WRITE (W,2050)
      LINES=LINES+4
 2050 FORMAT (/1X,125("-")///)
  230 CONTINUE
C
C  PRINT THE RESULTS FOR TYPE 2 IF ANY
C
      IF (NT(2).EQ.0.OR.INREP.EQ.0) GO TO 260
      I1=2
      IF (LINES.LT.LINPAG-11) GO TO 240
      CALL HEADER(2)
      LINES=10
  240 WRITE (W,2110) I1
      LINES=LINES+7
 2120 FORMAT (/T24,"WASTE"/T24,"FIRST",T32,"YEARLY",T121,"TOTAL"/
     1 T11."SOURCE",T24,"YEAR",T32,"GROWTH",T121,"WASTE"/T3."LABEL",
     2 T12,"NAME",T23,"TONNES",T34,"%".T121,"TONNES"/)
      WRITE (W,2120)
      DO 245 J=1,NUMSRC
      IF (ITYPE(J).NE.2) GO TO 245
      IF (LINES.LT.LINPAG) GO TO 250
      CALL HEADER(2)
      WRITE (W,2115) I1
      WRITE (W,2120)
      LINES=10
  250 WRITE (W,2130) SRCIDN(J),SRCNAM(J),AMUNT1(J),GROWTH(J),
     1 SRCTOT(J)
 2130 FORMAT (1X,I5,3X,A8,2X.F10.0,F7.2,T117,F10.0)
      LINES=LINES+1
  245 CONTINUE
      WRITE (W,2050)
      LINES=LINES+5
  260 CONTINUE
C
C  PRINT RESULTS FOR TYPE 3 IF ANY ARE PRESENT
C
      I1=3
      IF (NT(3).EQ.0) GO TO 170
      IF (LINES+10.GT.LINPAG.AND.INREP.GT.0) CALL HEADER(2)
      IF (LINES+10.GT.LINPAG) LINES=10
      WRITE (W,2110) I1
      WRITE (W,2069)
 2069 FORMAT (/T121,"TOTAL"/T10,"SOURCE",1X,4(14X,"AMOUNT",3X),T121,
     1 "WASTE"/1X,"LABEL",T11,"NAME".2X,4(4X,"YEAR",6X,"TONNES",3X),
     2 T121."TONNES"/)
      LINES=LINES+7
      DO 135 J=1,NUMSRC
      IF (ITYPE(J).NE.3) GO TO 135
      IF(NEN3(J).EQ.2)SRCTOT(J) =(AM3(1,J)+AM3(2,J))*RUNYRS/2.
      IF(NEN3(J).EQ.3)SRCTOT(J)=(AM3(1,J)+AM3(2,J))
     1                           *(IYR3(2,J)-IYR3(1,J)+1)/2.
     2           +(AM3(2,J)+AM3(3,J))*(IYR3(3,J)-IYR3(2,J)+1)/2.
     3                             -AM3(2,J)
      IF(NEN3(J).EQ.4)SRCTOT(J)=(AM3(1,J)+AM3(2,J))
     1                               *(IYR3(2,J)-IYR3(1,J)+1)/2.          00054500
     2          +(AM3(2,J)+AM3(3,J))*(IYR3(3,J)-IYR3(2,J)+1)/2.          00054600
     3          +(AM3(3,J)+AM3(4,J))*(IYR3(4,J)-IYR3(3,J)+1)/2.          00054700
     4                          -AM3(2,J)-AM3(3,J)                      00054800
      ALWAST = ALWAST+SRCTOT(J)                                         00054900
      IF (LINES.LT.LINPAG.OR.INREP.EQ.0) GO TO 190                      00055000
      CALL HEADER(2)                                                    00055100
      WRITE (W,2115) I1                                                 00055200
      WRITE (W,2069)                                                    00055300
      LINES=10                                                          00055400
  190 IF(INREP.GT.0) WRITE(W,2070) SRCIDN(J),SRCNAM(J),                 00055500
     1                             (IYR3(I,J),AM3(I,J),I=1,4),          00055600
     2                             SRCTOT(J)                            00055700
 2070 FORMAT(1X,I4,3X,A8,4(4X,I4,2X,F10.0,3X),T117,F10.0)               00055800
      LINES=LINES+1                                                     00055900
  135 CONTINUE                                                          00056000
      WRITE (W,2050)                                                    00056100
      LINES=LINES+5                                                     00056200
  170 CONTINUE                                                          00056300
C                                                                       00056400
C  PRINT THE RESULTS FOR TYPE 4 IF ANY ARE PRESENT                      00056500
C                                                                       00056600
      IF (NT(4).EQ.0.OR.INREP.EQ.0) GO TO 295                           00056700
      I1=4                                                              00056800
      IF (LINES.LT.LINPAG-11) GO TO 270                                 00056900
      CALL HEADER(2)                                                    00057000
      LINES=10                                                          00057100
  270 WRITE (W,2110) I1                                                 00057200
      LINES=LINES+7                                                     00057300
      WRITE (W,2140)                                                    00057400
 2140 FORMAT (/T23,"WASTE",T37,"WASTE",T62,"POPULATION",T121,           00057500
    1 "TOTAL"/T10,"SOURCE",T21,"FIRST YEAR",T37,"GROWTH",T49,           00057600
    2 "POPULATION",T64,"GROWTH",T121,"WASTE"/1X,"LABEL",T11,            00057700
    3 "NAME",T20,"TONNES/CAPITA",T36,"% PER YEAR",T49,"FIRST YEAR",     00057800
    4 T62,"% PER YEAR",T121,"TONNES"/)                                  00057900
      DO 280 J=1,NUMSRC                                                 00058000
      IF (ITYPE(J).NE.4) GO TO 280                                      00058100
      IF (LINES.LT.LINPAG) GO TO 290                                    00058200
      CALL HEADER(2)                                                    00058300
      WRITE (W,2115) I1                                                 00058400
      WRITE(W,2140)                                                     00058500
      LINES=10                                                          00058600
  290 WRITE (W,2150) SRCIDN(J),SRCNAM(J),TONYR1(J),TONGRO(J),           00058700
    1 POPYR1(J),POPGRO(J),SRCTOT(J)                                     00058800
 2150 FORMAT (1X,I4,3X,A8,F12.3,F14.2,F16.0,F10.2,T117,F10.0)           00058900
      LINES=LINES+1                                                     00059000
  280 CONTINUE                                                          00059100
      WRITE (W,2050)                                                    00059200
      LINES=LINES+5                                                     00059300
  295 CONTINUE                                                          00059400
C                                                                       00059500
C  PRINT RESULTS FOR TYPE 5 IF THERE ARE ANY                            00059600
C                                                                       00059700
      I1=5                                                              00059800
      IF (NT(5).EQ.0) GO TO 180                                         00059900
      IF (LINES+10.GT.LINPAG.AND.INREP.GT.0) CALL HEADER(2)             00060000
      IF (LINES+10.GT.LINPAG) LINES=10                                  00060100
```

```
       WRITE (W,2110) I1
       WRITE (W,2092)
       LINES=LINES+7
2092   FORMAT (/T18,"I",4(16X,"WASTE",2X,"I")/T18,"I",
     1 4(17X,"PER",3X,"I"),T121,"TOTAL"/T10,"SOURCE",T18,"I",
     2 4(6X,"POPULA-",3X,"CAPITA",1X,"I"),T121,"WASTE"/1X,
     3 "LABEL",T11,"NAME",T18,"I",4(1X,"YEAR",1X,"TION",6X,"TONNES",
     4 1X,"I"),T121,"TONNES"/T18,"I",4(23X,"I") )
       DO 145 J=1,NUMSRC
       IF (ITYPE(J).NE.5) GO TO 145
       NE=NEN3(J)
       DO 160 K=1,NE
160    AMOUNT(K) =TONY3(K,J)*POPY3(K,J)
       IF(NEN3(J).EQ.2)SRCTOT(J)=(AMOUNT(1)+AMOUNT(2))*RUNYRS/2.
       IF(NEN3(J).EQ.3)SRCTOT(J)=(AMOUNT(1)+AMOUNT(2))
     1                                *(IYR3(2,J)-IYR3(1,J)+1)/2.
     2            +(AMOUNT(2)+AMOUNT(3))*(IYR3(3,J)-IYR3(2,J)+1)/2.
     3                            -AMOUNT(2)
       IF(NEN3(J).EQ.4)SRCTOT(J)=(AMOUNT(1)+AMOUNT(2))
     1                                 *(IYR3(2,J)-IYR3(1,J)+1)/2.
     2             +(AMOUNT(2)+AMOUNT(3))*(IYR3(3,J)-IYR3(2,J)+1)/2.
     3             +(AMOUNT(3)+AMOUNT(4))*(IYR3(4,J)-IYR3(3,J)+1)/2.
     4                           -AMOUNT(2)-AMOUNT(3)
      ALWAST=ALWAST+SRCTOT(J)
      IF (LINES.LT.LINPAG.OR.INREP.EQ.0) GO TO 300
      CALL HEADER(2)
      WRITE (W,2115) I1
      WRITE (W,2092)
      LINES=10
 300  IF(INREP.GT.0)WRITE(W,2093)SRCIDN(J),SRCNAM(J),
     1                     (IYR3(I,J),POPY3(I,J),TONY3(I,J),I=1,4)
     2                            ,SRCTOT(J)
2093   FORMAT(1X,I4,3X,A8,1X,"I",4(1X,I4,F9.0,F7.2,2X,"I"),
     1 T117,F10.0)
      LINES=LINES+1
 145  CONTINUE
      WRITE (W,2050)
 180  IF (LINES+5.GT.LINPAG) CALL HEADER(2)
      WRITE(W,2100)ALWAST
2100  FORMAT(///20X,
     2 "  TOTAL WASTE FOR THE AREA AND FOR THE WHOLE PERIOD, TONNES-",
     1 35("-"),T112,F15.1)
      RETURN
      END
      SUBROUTINE TRANFR
C
C       ***************************************************************
C       *                                                             *
C       *    THIS ROUTINE READS IN RAW DATA FOR POTENTIAL TRANSFER    *
C       *    STATIONS AND CALCULATES FIXED AND VARIABLE COSTS AND     *
C       *    OTHER QUANTITIES FOR THE MILP INPUT                      *
C       *    DIMENSIONS SET FOR MAXIMUM OF 20 TRANSFER STATIONS       *
C       *                                                             *
C       ***************************************************************
C
C      ***    INITIAL DECLARATION OF L VARIABLES AS REAL
C
       IMPLICIT REAL(L)
C
C      ***    DECLARATION OF VARIABLES FOR NAMES
C             AND TITLES
       REAL ASTRSK
       REAL TITLE,RUNNAM,VECNAM,SRCNAM,TRFNAM,PLTNAM,
     1        LFLNAM,RSCNAM,MKTNAM
C
       INTEGER R,W,T,D,LINES
      INTEGER ENDPAY,ENDPX
       INTEGER TOTERR,ERFLAG,RUNYRS,
     1           RUNBEG,RUNEND,INREP,OUTREP,TWRITE,DWRITE,LANDCO,
     2            SRCIDN,TRFIDN,PLTIDN,LFLIDN,MKTIDN,
     3           ORIGID,DESTID,TRSTAT,PLSTAT,LFSTAT,STATJ
C
C      *** GENERAL COMMON BLOCK
C
       COMMON/GENDAT/ASTRSK(15),TITLE(5),RUNNAM(5),
     1                 LTOLFL,LNUMTR,LNUMPL,LNUMLF,LCAPTL,RATE,
     2                 R,W,T,D,TOTERR,ERFLAG,DAT,
     3                 RUNBEG,RUNEND,RUNYRS,INREP,OUTREP,TWRITE,DWRITE,
     4                 LANDCO,NUMVEC,NUMSRC,NUMTRF,NUMPLT,NUMLFL,NRESRC,
     5                 NUMMKT(9),NPAGE,LINPAG
C
C
C      ***    TRANSFER STATIONS COMMON BLOCK
C
       COMMON/TRFDAT/TRFNAM(20),TRFLLT(20),TRFULT(20),
     1                 TRFFCP(20),TRFVCP(20),TRFFOP(20),TRFVOP(20),
     2                 TRFIDN(20),TRSTAT(20)
C
C
      DIMENSION IYR(20,5,3),LMPPAY(20,5,3),THRU(20,5),IWAY(20)
      DIMENSION COSTC(20,5),COD(5),THR(5),FC(20,2),VC(20,2),RS(20,2)
      DIMENSION THRUD(20,5),COSTD(20,5),NCPE(20),NOPE(20),NYE(3),ALU(3)
      DIMENSION ENDPAY(20,5),AMTPAY(20,5),NLUMP(20),NOYES(2)
      DIMENSION TRFLL(20),TRFUL(20)
C
       REAL STATLB(3),CAPWAY(2),BL,NM,STA
       DATA STATLB/"NOT USED","  USED  ","  MODEL "/
       DATA CAPWAY/"AMORTIZE","LUMP PAY"/
       DATA NOYES/"NO  ","YES "/,BL/"        "/
C
    1 CONTINUE
```

```
      INCP=0
      INOP=0
      IWAY0=0
      IWAY1=0
C
      DO 300 J=1,NUMTRF
      READ(R,3005) TRFIDN(J),TRFNAM(J),TRSTAT(J),TRFLL(J),TRFUL(J),
     1  IWAY(J),NCPENT,NOPENT
      IF (NCPENT.GT.1) INCP=1
      IF (NOPENT.GT.1) INOP=1
      IF (IWAY(J).EQ.0) IWAY0=1
      IF (IWAY(J).EQ.1) IWAY1=1
      NCPE(J)=NCPENT
      NOPE(J)=NOPENT
 3005 FORMAT(I3,A8,3X,I1,5X,2F10.0,3(4X,I1))
      IF(TRSTAT(J).LT.0.OR.TRSTAT(J).GT.2) CALL ERROR(301)
       STATJ = TRSTAT(J)
      IF(TRFLL(J).GT.TRFUL(J)) CALL ERROR(302)
C
C     ***   READ AND CALCULATE CAPITAL ENTRIES FOR ALL SIZES
C     ***   OF TRANSFER STATIONS
C
      DO 110 K=1,NCPENT
      IF(IWAY(J).NE.0)GO TO 105
      READ(R,3020)IDNUM,THRU(J,K),ENDPAY(J,K),AMTPAY(J,K)
 3020 FORMAT(I3,7X,F10.0,6X,I4,F10.0)
      IF(IDNUM.NE.TRFIDN(J)) CALL ERROR(311)
      NYRS=ENDPAY(J,K)-RUNBEG
      COSTC(J,K)=AMTPAY(J,K)*(1.+USPWF(RATE,NYRS))
      DO 245 I=1,3
      IYR(J,K,I)=0
  245 LMPPAY(J,K,I)=0.
      GO TO 109
  105 READ(R,3030)IDNUM,NLUMP(J),THRU(J,K),(IYR(J,K,I),
     1 LMPPAY(J,K,I),I=1,3)
 3030 FORMAT(I3,6X,I1,F10.0,3(6X,I4,F10.0))
      IF(IDNUM.NE.TRFIDN(J)) CALL ERROR(311)
      COSTC(J,K)=0
       NLP=NLUMP(J)
       DO 107 I = 1,NLP
  107  COSTC(J,K)=COSTC(J,K)+LMPPAY(J,K,I)/(1.+RATE)**(IYR(J,K,I)-
     1 RUNBEG)
  109 COO(K)=COSTC(J,K)
      THR(K)=THRU(J,K)
  110 CONTINUE
      CALL REGRES(NCPENT,THR,COO,FCAP,VCAP,RSQCAP)
      FC(J,1)=FCAP
      VC(J,1)=VCAP
      RS(J,1)=RSQCAP
C
C                   CONVERT FOR ENTRY TO MILP
C
      TRFFCP(J)=FCAP
      TRFVCP(J)=VCAP/RUNYRS
C
C     ***   NOW OPERATING COSTS
C
```

```
      DO 210 K=1,NOPENT
      READ(R,3040)IDNUM,THRUO(J,K),COSTO(J,K)
      THR(K)=THRUO(J,K)
      COO(K)=COSTO(J,K)
 3040 FORMAT(I3,7X,2F10.0)
      IF(IDNUM.NE.TRFION(J)) CALL ERROR(311)
 210    CONTINUE
        CALL REGRES(NOPENT,THR,COO,FOPC,VOPC,RSQOP)
        FC(J,2)=FOPC
        VC(J,2)=VOPC
        RS(J,2)=RSQOP
C
C     ***   CONVERT TO FIGURES TO ENTER MILP
C
      FACTOR=USPWF(RATE,RUNYRS)
      TRFFOP(J) =FOPC*FACTOR
      TRFVOP(J) =VOPC*FACTOR/RUNYRS
      TRFLLT(J) =TRFLL(J)*RUNYRS
      TRFULT(J) =TRFUL(J)*RUNYRS
 300  CONTINUE
      IF (INREP.EQ.0) RETURN
C
C  PRINT THE RESULTS FOR AMORTIZED TRANSFER STATIONS
C
      IF (IWAYO.EQ.0) GO TO 155
      CALL HEADER(3)
      WRITE (W,3000)
 3000 FORMAT (/1X,"TRANSFER STATIONS WITH AMORTIZED CAPITAL COST"//
     1 T58,"ANNUAL",T70,"AMORTIZATION",T90,"ANNUAL",T104,"ANNUAL",
     2 T118, "ANNUAL"/T30,"THROUGHPUT LIMITS,TONNES",T57,"CAPACITY",
     3 T69,"LAST",T77,"ANNUAL",T90,"CAPITAL",T102,"THROUGHPUT",
     4 T116,"OPERATING"/1X,"LABEL",T11,"NAME",T20,"STATUS",T34,
     5 "LEAST",T47,"MOST",T58,"TONNES",T69,"YEAR",T76,"PAYMENT,$",
     6 T91,"COST,$",T104,"TONNES",T118,"COST,$"/)
3010  FORMAT (1X)
      LINES=16
      DO 150 J=1,NUMTRF
      IF (IWAY(J).NE.0) GO TO 150
      NCPENT=NCPE(J)
      NOPENT=NOPE(J)
      STATJ=TRSTAT(J)
      IF (NCPENT.GT.1.OR.NOPENT.GT.1) WRITE (W,3010)
      IF (NCPENT.GT.1.OR.NOPENT.GT.1) LINES=LINES+1
      NEND=MAX0(NCPENT,NOPENT)
3025  FORMAT (1X,I4,3X,A8,2X,A8,1X,2F12.0,F14.0,3X,I4,F12.0,3F13.0)
      DO 140 N=1,NEND
      NM=8L
      STA=8L
      ALLT=0.0
      AULT=0.0
      THRUX=0.0
      ENDPX=0
      AMTPX=0.
      COSTCX=0.0
      THRUOX=0.
      COSTOX=0.
      IF (N.GT.1) GO TO 120
      NM=TRENAM(J)
      STA=STATLB(STATJ+1)
      ALLT=TRELL(J)
      AULT=TREUL(J)
  120 IF (N.GT.NCPENT) GO TO 130
      THRUX=THRU(J,N)
      ENDPX=ENDPAY(J,N)
      AMTPX=AMTPAY(J,N)
      COSTCX=COSTC(J,N)
  130 IF (N.GT.NOPENT) GO TO 180
      THRUOX=THRUO(J,N)
      COSTOX=COSTO(J,N)
  180 WRITE (W,3025) TREIDN(J),NM,STA,ALLT,AULT,THRUX,ENDPX,AMTPX,
     1 COSTCX,THRUOX,COSTOX
      LINES=LINES+1
      IF (LINES.LT.LINPAG) GO TO 140
      CALL HEADER(3)
      WRITE (W,3000)
      LINES=16
  140 CONTINUE
      IF (NCPENT.GT.1.OR.NOPENT.GT.1) WRITE (W,3010)
      IF (NCPENT.GT.1.OR.NOPENT.GT.1) LINES=LINES+1
  150 CONTINUE
  155 IF (IWAYO.EQ.0) CALL HEADER(3)
      IF (IWAYO.EQ.0) LINES=10
      IF (IWAY1.EQ.0) GO TO 235
C
C  PRINT THE RESULTS FOR STATIONS WITH LUMP PAYMENTS
C  FOR CAPITAL COSTS
C
 3034 FORMAT (///1X,"TRANSFER STATIONS WITH LUMP PAYMENTS OF CAPITAL COS
     1T"/)
 3035 FORMAT (/T47,"ANNUAL",T104,"ANNUAL",T114,"ANNUAL",T126,
     1 "ANNUAL"/T26,"THROUGHPUT,TONNES",T46,"CAPACITY",T62,
     2 "LUMP PAYMENTS OF CAPITAL COST",T103,"CAPITAL",T112,
     3 "THROUGHPUT",T124,"OPERATING"/1X,"LABEL",T9,"NAME",T17,
     4 "STATUS",T28,"LEAST",T38,"MOST",T47,"TONNES",T56,
     5 3("YEAR   -    $      "),T104,"COST,$",T114,"TONNES",T126,
     6 "COST,$"/)
 3045 FORMAT (1X,I4,1X,A8,1X,A8,2F10.0,F11.0,3(1X,I4,F10.0),3F11.0)
      IF (LINES.LT.LINPAG-12) GO TO 215
      CALL HEADER(3)
      LINES=10
  215 WRITE (W,3034)
  220 WRITE (W,3035)
      LINES=LINES+8
      DO 230 J=1,NUMTRF
      IF (IWAY(J).NE.1) GO TO 230
      NCPENT=NCPE(J)
      NOPENT=NOPE(J)
      STATJ=TRSTAT(J)
      IF (NCPENT.GT.1.OR.NOPENT.GT.1) WRITE (W,3010)
      IF (NCPENT.GT.1.OR.NOPENT.GT.1) LINES=LINES+1
      NEND=MAX0(NCPENT,NOPENT)
      DO 240 N=1,NEND
      NM=BL
      STA=BL
      ALLT=0.                                                           00087200
      AULT=0.                                                           00087300
      THRUX=0.                                                          00087400
      NYE(1)=0                                                          00087500
      NYE(2)=0                                                          00087600
      NYE(3)=0                                                          00087700
      ALU(1)=0.                                                         00087800
      ALU(2)=0.                                                         00087900
      ALU(3)=0.                                                         00088000
      COSTCX=0.                                                         00088100
      THRUOX=0.                                                         00088200
      COSTOX=0.                                                         00088300
      IF (N.GT.1) GO TO 250                                             00088400
      NM=TRFNAM(J)                                                      00088500
      STA=STATLB(STATJ+1)                                               00088600
      ALLT=TRFLL(J)                                                     00088700
      AULT=TRFUL(J)                                                     00088800
  250 IF (N.GT.NCPENT) GO TO 265                                        00088900
      THRUX=THRU(J,N)                                                   00089000
      DO 260 I=1,3                                                      00089100
      NYE(I)=IYR(J,N,I)                                                 00089200
  260 ALU(I)=LMPPAY(J,N,I)                                              00089300
  265 COSTCX=COSTC(J,N)                                                 00089400
      IF (N.GT.NOPENT) GO TO 270                                        00089500
      THRUOX=THRU(J,N)                                                  00089600
      COSTOX=COSTO(J,N)                                                 00089700
  270 WRITE (W,3045) TRFIDN(J),NM,STA,ALLT,AULT,THRUX,(NYE(I),ALU(I),   00089800
     1 I=1,3),COSTCX,THRUOX,COSTOX                                      00089900
      IF (LINES.LT.LINPAG) GO TO 240                                    00090000
      CALL HEADER(3)                                                    00090100
      WRITE (W,3034)                                                    00090200
      WRITE (W,3035)                                                    00090300
      LINES=16                                                          00090400
  240 CONTINUE                                                          00090500
      IF (NCPENT.GT.1.OR.NOPENT.GT.1) WRITE (W,3010)                    00090600
      IF (NCPENT.GT.1.OR.NOPENT.GT.1) LINES=LINES+1                     00090700
  230 CONTINUE                                                          00090800
  235 IF (INCP.EQ.0.AND.INOP.EQ.0) RETURN                               00090900
      IF (LINES.LT.LINPAG-9) GO TO 325                                  00091000
      CALL HEADER(3)                                                    00091100
      LINES=10                                                          00091200
  325 WRITE (W,3050)                                                    00091300
 3050 FORMAT (//1X,"TRANSFER STATIONS - RESULTS OF REGRESSION"//1X,     00091400
     1 "LABEL",T10,"NAME",T22,"INTERCEPT",T37,"SLOPE",T48,"RSQUARE",    00091500
     2 T58,"CAPITAL COST VS. CAPACITY",3X,"OPERATIONAL COST VS.THROUGHPU00091600
     3T"/)                                                              00091700
      LINES=LINES+5                                                     00091800
      DO 310 J=1,NUMTRF                                                 00091900
      IF (LINES.LT.LINPAG) GO TO 330                                    00092000
      CALL HEADER(3)                                                    00092100
      WRITE (W,3050)                                                    00092200
  330 NOPENT=NOPE(J)                                                    00092300
      NCPENT=NCPE(J)                                                    00092400
      IF (NOPENT.EQ.1.AND.NCPENT.EQ.1) GO TO 310                        00092500
      IF (NOPENT.EQ.1) GO TO 320                                        00092600
      WRITE (W,3060) TRFIDN(J),TRFNAM(J),FC(J,2),VC(J,2),RS(J,2)        00092700
      LINES=LINES+1                                                     00092800
 3060 FORMAT (1X,I4,2X,A8,F15.1,F13.5,F10.3,T68,"X")
  320 IF (NCPENT.EQ.1) GO TO 310
      WRITE (W,3070) TRFIDN(J),TRFNAM(J),FC(J,1),VC(J,1),RS(J,1)
 3070 FORMAT (1X,I4,2X,A8,F15.1,F13.5,F10.3,T96,"X")
      LINES=LINES+1
  310 CONTINUE
      RETURN
      END
```

```
      REAL FUNCTION USPWF(R,N)
C
C     ******************************************************************
C     *                                                                *
C     *  FUNCTION TO COMPUTE UNIFORM SERIES PRESENT WORTH FACTOR       *
C     *                                                                *
C     ******************************************************************
C
       IF(R.NE.0) GO TO 10
       USPWF=N
       RETURN
   10 USPWF =((1.+R)**N-1.)/(R*(1.+R)**N)
      RETURN
      END
      SUBROUTINE REGRES (N,X;Y,F,V,RSQ)
C
C     ***********************************************************************
C     *                                                                     *
C     *     THIS ROUTINE CALCULATES REGRESSION FIGURES.                     *
C     *     IF THERE IS JUST ONE OBSERVATION, F IS SET TO THIS COST.        *
C     *     IF F IS FOUND BY SIMPLE REGRESSION TO BE NEGATIVE, THEN         *
C     *     F IS SET TO ZERO AND REGRESSION FORCED THROUGH ORIGIN.          *
C     *     F IS INTERCEPT                                                  *
C     *     V IS SLOPE                                                      *
C     *                                                                     *
C     ***********************************************************************
C
      DIMENSION X(5), Y(5)
      IF(N.GT.1) GO TO 50
      F=Y(1)
      V=0.
      RSQ=0.
      RETURN
 50   CONTINUE
      SX=0.
      SXX=0.
      SY=0.
      SYY=0.
      SXY=0.
      DO 100 J=1,N
      SX=SX+X(J)
      SXX=SXX+X(J)*X(J)
      SY=SY+Y(J)
      SYY=SYY+Y(J)*Y(J)
      SXY=SXY+X(J)*Y(J)
 100  CONTINUE
      VARNX=SXX-SX*SX/N
      VARNY=SYY-SY*SY/N
      VARNXY=SXY-SX*SY/N
      V=VARNXY/VARNX
      F=(SY-V*SX)/N
      RSQ=V*V*VARNX/VARNY
      IF(F.GE.0.) RETURN
      F=0.
      V=SXY/SXX
      RSC=V*SXY/SYY
      RETURN
      END
      SUBROUTINE PLANTS
C
C       ***************************************************************
C       *                                                             *
C       *    THIS ROUTINE READS IN RAW DATA FOR POTENTIAL PROCESSING  *
C       *    PLANTS AND CALCULATES FIXED AND VARIABLE COSTS AND       *
C       *    OTHER QUANTITIES FOR THE MILP INPUT                      *
C       *                                                             *
C       ***************************************************************
C
C      ***    INITIAL DECLARATION OF L VARIABLES AS REAL
C
       IMPLICIT REAL(L)
C
C      ***    DECLARATION OF VARIABLES FOR NAMES AND
C             TITLES
       REAL ASTRSK
       REAL TITLE,RUNNAM,VECNAM,SRCNAM,TRFNAM,PLTNAM,
     1         LFLNAM,RSCNAM,MKTNAM
C
       INTEGER R,W,T,D,LINES
      INTEGER ENDPAY,ENDPX
       INTEGER TOTERR,ERFLAG,RUNYRS,
     1             RUNBEG,RUNEND,INREP,OUTREP,TWRITE,DWRITE,LANDCD,
     2              SRCIDN,TRFIDN,PLTIDN,LFLIDN,MKTIDN,
     3             ORIGID,DESTID,TRSTAT,PLSTAT,LFSTAT,STATJ
C
C      *** GENERAL COMMON BLOCK
C
       COMMON/GENDAT/ASTRSK(15),TITLE(5),RUNNAM(5),
     1                 LTOLFL,LNUMTR,LNUMPL,LNUMLF,LCAPTL,RATE,
     2                 R,W,T,D,TOTERR,ERFLAG,DAT,
     3                 RUNBEG,RUNEND,RUNYRS,INREP,OUTREP,TWRITE,DWRITE,
     4                 LANDCD,NUMVEC,NUMSRC,NUMTRF,NUMPLT,NUMLFL,NRESRC,
     5                 NUMMKT(9),NPAGE,LINPAG
C
C
C      ***    COMMON FOR PLANTS
C
       COMMON/PLTDAT/PLTNAM(20),PLTLLT(20),PLTULT(20),
     1                 PLTFCP(20),PLTVCP(20),PLTFOP(20),PLTVOP(20),
     2                 PLTKEP(20),PLTRRN(20,9),
     3                 PLTIDN(20),PLSTAT(20)
C
C
      DIMENSION IYR(20,5,3),LMPPAY(20,5,3),THRU(20,5),IWAY(20)
      DIMENSION COSTC(20,5),COD(5),THR(5),FC(20,2),VC(20,2),RS(20,2)
      DIMENSION THRUD(20,5),COSTD(20,5),NCPE(20),NOPE(20),NYE(3),ALU(3)
      DIMENSION ENDPAY(20,5),AMTPAY(20,5),NLUMP(20),NOYES(2)
      DIMENSION PLTLL(20),PLTUL(20),PLTLOS(20)
C
       REAL STATLB(3),CAPWAY(2),BL,NM,STA
       DATA STATLB/"NOT USED","  USED  ","  MODEL "/
       DATA CAPWAY/"AMORTIZD","LUMP PAY"/
       DATA NOYES/"NO  ","YES "/,BL/"        "/
    1 CONTINUE
       INCP=0
       INPO=0
       IWAYO=0
       IWAY1=0
      DO 300 J=1,NUMPLT
      READ(R,3005) PLTION(J),PLTNAM(J),PLSTAT(J),PLTLL(J),PLTUL(J),
     1  IWAY(J),NCPENT,NOPENT
 3005 FORMAT(I3,A8,3X,I1,5X,2F10.0,3(4X,I1))
      IF(PLSTAT(J).LT.0.OR.PLSTAT(J).GT.2) CALL ERROR(401)
      IF(PLTLL(J).GT.PLTUL(J)) CALL ERROR(402)
      IF (NCPENT.GT.1) INCP=1
      IF (NOPENT.GT.1) INOP=1
      NCPE(J)=NCPENT
      NOPE(J)=NOPENT
      IF (IWAY(J).EQ.0) IWAYO=1
      IF (IWAY(J).EQ.1) IWAY1=1
      STATJ=PLSTAT(J)
C
C     ***  READ AND CALCULATE CAPITAL ENTRIES FOR ALL PLANT SIZES
C
      DO 110 K=1,NCPENT
      IF(IWAY(J).NE.0)GO TO 105
      READ(R,3020)IDNUM,THRU(J,K),ENDPAY(J,K),AMTPAY(J,K)
 3020 FORMAT(I3,7X,F10.0,6X,I4,F10.0)
      IF(IDNUM.NE.PLTION(J)) CALL ERROR(411)
      NYRS=ENDPAY(J,K)-RUNBEG
      COSTC(J,K)=AMTPAY(J,K)*(1.+USPWF(RATE,NYRS))
      DO 245 I=1,3
      IYR(J,K,I)=0
 245  LMPPAY(J,K,I)=0.
      GO TO 109
 105  READ(R,3030)IDNUM,NLUMP(J),THRU(J,K),(IYR(J,K,I),
     1 LMPPAY(J,K,I),I=1,3)
 3030 FORMAT(I3,6X,I1,F10.0,3(6X,I4,F10.0))
      IF(IDNUM.NE.PLTION(J)) CALL ERROR(411)
      COSTC(J,K)=0
      NLP=NLUMP(J)
      DO 107 I = 1,NLP
 107  COSTC(J,K)=COSTC(J,K)+LMPPAY(J,K,I)/(1.+RATE)**(IYR(J,K,I)
     1 - RUNBEG)
 109  COO(K)=COSTC(J,K)
      THR(K)=THRU(J,K)
 110  CONTINUE
      CALL REGRES(NCPENT,THR,COO,FCAP,VCAP,RSQCAP)
      FC(J,1)=FCAP
      VC(J,1)=VCAP
      RS(J,1)=RSQCAP
C
C                   CONVERT FOR ENTRY TO MILP
C
      PLTFCP(J)=FCAP
      PLTVCP(J)=VCAP/RUNYRS
C
C     ***  NOW OPERATING COSTS
C
      DO 210 K=1,NOPENT
      READ(R,3040)IDNUM,THRUO(J,K),COSTO(J,K)
      THR(K)=THRUO(J,K)
      COO(K)=COSTO(J,K)
 3040 FORMAT(I3,7X,2F10.0)
      IF(IDNUM.NE.PLTION(J)) CALL ERROR(411)
  210 CONTINUE
      CALL REGRES(NOPENT,THR,COO,FOPC,VOPC,RSQOP)
      FC(J,2)=FOPC
      VC(J,2)=VOPC
      RS(J,2)=RSQOP
      READ (R,3019) IDNUM,PLTLUS(J),(PLTRRN(J,M),M=1,9)
 3019 FORMAT (I3,7X,10F5.2)
      IF (IDNUM.NE.PLTION(J)) CALL ERROR(411)
C
C     ***   CONVERT TO FIGURES TO ENTER MILP
C
      FACTOR=USPWF(RATE,RUNYRS)
      PLTFOP(J) =FOPC*FACTOR
      PLTVOP(J) =VOPC*FACTOR/RUNYRS
      PLTLLT(J) =PLTLL(J)*RUNYRS
      PLTULT(J) =PLTUL(J)*RUNYRS
  300 CONTINUE
      IF (INREP.EQ.0) RETURN
C
C  PRINT THE RESULTS FOR AMORTIZED PLANTS
C
      IF (IWAYO.EQ.0) GO TO 155
      CALL HEADER(4)
      WRITE (W,3000)
 3000 FORMAT (/1X,"PLANTS WITH AMORTIZED CAPITAL COST"//
     1 T58,"ANNUAL",T70,"AMORTIZATION",T90,"ANNUAL",T104,"ANNUAL",
     2 T118, "ANNUAL"/T39,"THROUGHPUT LIMITS,TONNES",T57,"CAPACITY",
     3 T69,"LAST",T77,"ANNUAL",T90,"CAPITAL",T102,"THROUGHPUT",
     4 T116,"OPERATING"/1X,"LABEL",T11,"NAME",T20,"STATUS",T34,
     5 "LEAST",T47,"MOST",T58,"TONNES",T69,"YEAR",T76,"PAYMENT,$",
     6 T91,"COST,$",T104,"TONNES",T118,"COST,$"/)
3010  FORMAT (1X)
      LINES=16
      DO 150 J=1,NUMPLT
      IF (IWAY(J).NE.0) GO TO 150
      NCPENT=NCPE(J)
      NOPENT=NOPE(J)
      STATJ=PLSTAT(J)
      IF (NCPENT.GT.1.OR.NOPENT.GT.1) WRITE (W,3010)
      IF (NCPENT.GT.1.OR.NOPENT.GT.1) LINES=LINES+1
      NEND=MAX0(NCPENT,NOPENT)
3025  FORMAT (1X,I4,3X,A8,2X,A8,1X,2F12.0,F14.0,3X,I4,F12.0,3F13.0)
      DO 140 N=1,NEND
      NM=BL
      STA=BL
      ALLT=0.0
      AULT=0.0
      THRUX=0.0
      ENDPX=0
      AMTPX=0.
      COSTCX=0.0
      THRUOX=0.
      COSTOX=0.
      IF (N.GT.1) GO TO 120
      NM=PLTNAM(J)
      STA=STATLB(STATJ+1)
      ALLT=PLTLL(J)
      AULT=PLTUL(J)
 120  IF (N.GT.NCPENT) GO TO 130
      THRUX=THRU(J,N)
      ENDPX=ENDPAY(J,N)
      AMTPX=AMTPAY(J,N)
      COSTCX=COSTC(J,N)
 130  IF (N.GT.NOPENT) GO TO 180
      THRUOX=THRUO(J,N)
      COSTOX=COSTO(J,N)
 180  WRITE (W,3025) PLTIDN(J),NM,STA,ALLT,AULT,THRUX,ENDPX,AMTPX,
     1 COSTCX,THRUOX,COSTOX
      LINES=LINES+1
      IF (LINES.LT.LINPAG) GO TO 140
      CALL HEADER(4)
      WRITE (W,3000)
      LINES=16
 140  CONTINUE
      IF (NCPENT.GT.1.OR.NOPENT.GT.1) WRITE (W,3010)
      IF (NCPENT.GT.1.OR.NOPENT.GT.1) LINES=LINES+1
 150  CONTINUE
 155  IF (IWAYO.EQ.0) CALL HEADER(4)
      IF (IWAYO.EQ.0) LINES=10
       IF (IWAY1.EQ.0) GO TO 238
C
C  PRINT THE RESULTS FOR STATIONS WITH LUMP PAYMENTS
C  FOR CAPITAL COSTS
C
 3034 FORMAT (///1X,"PLANTS WITH LUMP PAYMENTS OF CAPITAL COST"/)
 3035 FORMAT (/T47,"ANNUAL",T104,"ANNUAL",T114,"ANNUAL",T126,
     1 "ANNUAL"/T26,"THROUGHPUT, TONNES",T46,"CAPACITY",T62,
     2 "LUMP PAYMENTS OF CAPITAL COST",T103,"CAPITAL",T112,
     3 "THROUGHPUT",T124,"OPERATING"/1X,"LABEL",T9,"NAME",T17,
     4 "STATUS",T28,"LEAST",T38,"MOST",T47,"TONNES",T56,
     5 3("YEAR   -    $      "),T104,"COST,$",T114,"TONNES",T126,
     6 "COST,$"/)
 3045 FORMAT (1X,I4,1X,A8,1X,A8,2F10.0,F11.0,3(1X,I4,F10.0),3F11.0)
      IF (LINES.LT.LINPAG-12) GO TO 215
      CALL HEADER(4)
      LINES=10
  215 WRITE (W,3034)
  220 WRITE (W,3035)
      LINES=LINES+8
      DO 230 J=1,NUMPLT
      IF (IWAY(J).NE.1) GO TO 230
      NCPENT=NCPE(J)
      NOPENT=NOPE(J)
      STATJ=PLSTAT(J)
      IF (NCPENT.GT.1.OR.NOPENT.GT.1) WRITE (W,3010)
      IF (NCPENT.GT.1.OR.NOPENT.GT.1) LINES=LINES+1
      NEND=MAX0(NCPENT,NOPENT)
      DO 240 N=1,NEND
      NM=BL
      STA=BL
      ALLT=0.
      AULT=0.
      THRUX=0.
      NYE(1)=0
      NYE(2)=0
      NYE(3)=0
      ALU(1)=0.
      ALU(2)=0.
      ALU(3)=0.
      COSTCX=0.
      THRUOX=0.
      COSTOX=0.
      IF (N.GT.1) GO TO 250
      NM=PLTNAM(J)
      STA=STATLB(STATJ+1)
      ALLT=PLTLL(J)
      AULT=PLTUL(J)
 250  IF (N.GT.NCPENT) GO TO 268
      THRUX=THRU(J,N)
      DO 260 I=1,3
      NYE(I)=IYR(J,N,I)
 260  ALU(I)=LMPPAY(J,N,I)
 268  COSTCX=COSTC(J,N)
      IF (N.GT.NOPENT) GO TO 270
      THRUOX=THRU(J,N)
      COSTOX=COSTU(J,N)
 270  WRITE (W,3045) PLTION(J),NM,STA,ALLT,AULT,THRUX,(NYE(I),ALU(I),
     1 I=1,3),COSTCX,THRUOX,COSTOX
      IF (LINES.LT.LINPAG) GO TO 240
      CALL HEADER(4)
      WRITE (W,3034)
      WRITE (W,3035)
      LINES=16
 240  CONTINUE
      IF (NCPENT.GT.1.OR.NOPENT.GT.1) WRITE (W,3010)
      IF (NCPENT.GT.1.OR.NOPENT.GT.1) LINES=LINES+1
 230  CONTINUE
 238  CONTINUE
      IF (LINES.GT.LINPAG-14) GO TO 255
      WRITE (W,3110) (I,I=1,9)
3110 FORMAT (///T9,"PLANT",T18,"% WASTE  ",15("-"),"% WASTE",
    1 " RECOVERED AS RESOURCE",15("-")/1X,"LABEL  NAME",T19,
    2 "LOST",9(6X,I1)/)
  255 LINES=LINES+5
  325 CONTINUE
      DO 265 J=1,NUMPLT
      IF (LINES.LT.LINPAG-8) GO TO 275
      CALL HEADER(4)
      LINES=11
      WRITE (W,3110) (I,I=1,9)
  275 WRITE (W,3120) PLTION(J),PLTNAM(J),PLTLOS(J),(PLTRRN(J,I),
     1 I=1,9)
      LINES=LINES+1
3120 FORMAT (1X,I4,2X,A8,F8.2,1X,9F7.2)
      PLTKEP(J)=-(1-PLTLOS(J)/100.)
      IF (NRESRC.EQ.0) GO TO 265
      DO 11 M=1,NRESRC
 11   PLTRRN(J,M)=-PLTRRN(J,M)/100
```

```
265   CONTINUE
235   IF (INCP.EQ.0.AND.INOP.EQ.0) RETURN
      IF (LINES.LT.LINPAG-9) GO TO 335
      CALL HEADER(4)
      LINES=10
  335 WRITE (W,3050)
 3050 FORMAT (//1X,"PLANTS - RESULTS OF REGRESSION"//1X,
     1 "LABEL",T10,"NAME",T22,"INTERCEPT",T37,"SLOPE",T48,"RSQUARE",
     2 T58,"CAPITAL COST VS. CAPACITY",3X,"OPERATIONAL COST VS.THROUGHPU
     3T"/)
      LINES=LINES+5
      DO 310 J=1,NUMPLT
      IF (LINES.LT.LINPAG) GO TO 330
      CALL HEADER(4)
      WRITE (W,3050)
  330 NOPENT=NOPE(J)
      NCPENT=NCPE(J)
      IF (NOPENT.EQ.1.AND.NCPENT.EQ.1) GO TO 310
      IF (NOPENT.EQ.1) GO TO 320
      WRITE (W,3050) PLTION(J),PLTNAM(J),FC(J,2),VC(J,2),RS(J,2)
      LINES=LINES+1
 3060 FORMAT (1X,I4,2X,A8,F15.1,F13.5,F10.3,T68,"X")
  320 IF (NCPENT.EQ.1) GO TO 310
      WRITE (W,3070) PLTION(J),PLTNAM(J),FC(J,1),VC(J,1),RS(J,1)
 3070 FORMAT (1X,I4,2X,A8,F15.1,F13.5,F10.3,T96,"X")
      LINES=LINES+1
  310 CONTINUE
      RETURN
      END
      SUBROUTINE LANDFL
C
C       ****************************************************************
C       *                                                              *
C       *   THIS ROUTINE READS IN RAW DATA FOR POTENTIAL LANDFILL      *
C       *   SITES AND CALCULATES FIXED AND VARIABLE COSTS AND          *
C       *   OTHER QUANTITIES FOR THE MILP INPUT                        *
C       *                                                              *
C       ****************************************************************
C      ***   INITIAL DECLARATION OF L VARIABLES AS REAL
C
       IMPLICIT REAL(L)
C
C      ***   DECLARATION OF VARIABLES FOR NAMES AND
C            TITLES
       REAL ASTRSK
       REAL TITLE,RUNNAM,VECNAM,SRCNAM,TRFNAM,PLTNAM,
     1         LFLNAM,RSCNAM,MKTNAM
C
       INTEGER R,W,T,D
      INTEGER ENDPAY,LINES,ENDPX
       INTEGER TOTERR,ERFLAG,RUNYRS,
     1             RUNBEG,RUNEND,INREP,OUTREP,TWRITE,DWRITE,LANDCD,
     2              SRCIDN,TRFIDN,PLTIDN,LFLIDN,MKTIDN,
     3             ORIGID,DESTID,TRSTAT,PLSTAT,LFSTAT,STATJ
C
C      *** GENERAL COMMON BLOCK
C
       COMMON/GENDAT/ASTRSK(15),TITLE(5),RUNNAM(5),
     1                  LTOLFL,LNUMTR,LNUMPL,LNUMLF,LCAPTL,RATE,
     2                  R,W,T,D,TOTERR,ERFLAG,DAT,
     3                  RUNBEG,RUNEND,RUNYRS,INREP,OUTREP,TWRITE,DWRITE,
     4                  LANDCD,NUMVEC,NUMSRC,NUMTRF,NUMPLT,NUMLFL,NRESRC,
     5                  NUMMKT(9),NPAGE,LINPAG
C
C
C      ***   COMMON FOR LANDFILLS
C
       COMMON/LFLDAT/LFLNAM(20),LFLLLT(20),LFLULT(20),
     1                  LFLFCP(20),LFLVCP(20),LFLFOP(20),LFLVOP(20),
     2                  LFLIDN(20),LFSTAT(20)
C
C
      DIMENSION IYR(20,5,3),LMPPAY(20,5,3),THRU(20,5),IWAY(20)
      DIMENSION COSTC(20,5),COO(5),THR(5),FC(20,2),VC(20,2),RS(20,2)
      DIMENSION THRUO(20,5),COSTO(20,5),NCPE(20),NOPE(20),LFMOST(20)
      DIMENSION ENDPAY(20,5),AMTPAY(20,5),NLUMP(20),NOYES(2),NYE(3)
      DIMENSION ALU(3),LFLLL(20),LFLUL(20)
C
C
       REAL STATLB(3),CAPWAY(2),BL,NM.STA
       DATA STATLB/"NOT USED","  USED  ","  MODEL "/
       DATA CAPWAY/"AMORTIZD","LUMP PAY"/
       DATA NOYES/"NO  ","YES "/,BL/"        "/
C
    1  CONTINUE
       INCP=0
      INOP=0
      IWAY0=0
      IWAY1=0
      DO 300 J=1,NUMLFL
      READ(R,3005) LFLIDN(J),LFLNAM(J),LFSTAT(J),LFLLL(J),LFLUL(J),
     1  IWAY(J),NCPENT,NOPENT,LFMUST(J)
 3005 FORMAT(I3,A8,3X,I1,5X,2F10.0,3(4X,I1),F10.0)
      IF (NCPENT.GT.1) INCP=1
      IF (NOPENT.GT.1) INOP=1
      NCPE(J)=NCPENT
      NOPE(J)=NOPENT
      IF (IWAY(J).EQ.0) IWAY0=1
      IF (IWAY(J).EQ.1) IWAY1=1
      IF(LFSTAT(J).LT.0.OR.LFSTAT(J).GT.2) CALL ERROR(501)
      IF(LFLLL(J).GT.LFLUL(J)) CALL ERROR(502)
       IF(LFLLL(J).GT.LFMUST(J)/RUNYRS) CALL ERROR(503)
       STATJ=LFSTAT(J)
C
C     ***   READ AND CALCULATE CAPITAL ENTRIES FOR ALL SIZES
C
      DO 110 K=1,NCPENT
      IF(IWAY(J).NE.0)GO TO 105
      READ(R,3020)IDNUM,THRU(J,K),ENDPAY(J,K),AMTPAY(J,K)
 3020 FORMAT(I3,7X,F10.0,6X,I4,F10.0)
      IF(IDNUM.NE.LFLIDN(J)) CALL ERROR(511)
      NYRS=ENDPAY(J,K)-RUNBEG
      COSTC(J,K)=AMTPAY(J,K)*(1.+USPWF(RATE,NYRS))
      DO 245 I=1,3
      IYR(J,K,I)=0
  245 LMPPAY(J,K,I)=0.
      GO TO 109
 105  READ(R,3030)IDNUM,NLUMP(J),THRU(J,K),(IYR(J,K,I),
     1 LMPPAY(J,K,I),I=1,3)
 3030 FORMAT(I3,6X,I1,F10.0,3(6X,I4,F10.0))
      IF(IDNUM.NE.LFLIDN(J)) CALL ERROR(511)
      COSTC(J,K)=0
      NLP=NLUMP(J)
       DO 107 I = 1,NLP
 107   COSTC(J,K)=COSTC(J,K)+LMPPAY(J,K,I)/(1.+RATE)**(IYR(J,K,I)
     1 - RUNBEG)
 109  COO(K)=COSTC(J,K)
      THR(K)=THRU(J,K)
 110  CONTINUE
      CALL REGRES(NCPENT,THR,COO,FCAP,VCAP,RSQCAP)
      FC(J,1)=FCAP
      VC(J,1)=VCAP
      RS(J,1)=RSQCAP
C
C                     CONVERT FOR ENTRY TO MILP
C
      LFLFCP(J)=FCAP
      LFLVCP(J)=VCAP/RUNYRS
C     ***   NOW OPERATING COSTS
      DO 210 K=1,NOPENT
      READ(R,3040)IDNUM,THRUO(J,K),COSTO(J,K)
 3040 FORMAT(I3,7X,2F10.0)
      THR(K)=THRUO(J,K)
      COO(K)=COSTO(J,K)
      IF(IDNUM.NE.LFLIDN(J)) CALL ERROR(511)
 210  CONTINUE
      CALL REGRES(NOPENT,THR,COO,FOPC,VOPC,RSQOP)
      FC(J,2)=FOPC
      VC(J,2)=VOPC
      RS(J,2)=RSQOP
C
C     ***   CONVERT TO FIGURES TO ENTER MILP
C
      FACTOR=USPWF(RATE,RUNYRS)
      LFLFOP(J) =FOPC*FACTOR
      LFLVOP(J) =VOPC*FACTOR/RUNYRS
      LFLLLT(J) =LFLLL(J)*RUNYRS
      LFLULT(J) =AMIN1(LFLUL(J)*RUNYRS,LFMOST(J))
 300  CONTINUE
      IF (INREP.EQ.0) RETURN
C
C  PRINT THE RESULTS FOR AMORTIZED LANDFILLS
C
      IF (IWAYO.EQ.0) GO TO 155
      CALL HEADER(5)
      WRITE (W,3000)
 3000 FORMAT (/1X,"LANDFILLS WITH AMORTIZED CAPITAL COST"//
     1 T59,"ANNUAL",T70,"AMORTIZATION",T90,"ANNUAL",T104,"ANNUAL",
     2 T118, "ANNUAL"/T30,"THROUGHPUT LIMITS,TONNES",T57,"CAPACITY",
     3 T69,"LAST",T77,"ANNUAL",T90,"CAPITAL",T102,"THROUGHPUT",
     4 T116,"OPERATING"/1X,"LABEL",T11,"NAME",T20,"STATUS",T34,
     5 "LEAST",T47,"MOST",T58,"TONNES",T69,"YEAR",T76,"PAYMENT,$",
     6 T91,"COST,$",T104,"TONNES",T118,"COST,$"/)
3010  FORMAT (1X)
      LINES=16
      DO 150 J=1,NUMLFL
      IF (IWAY(J).NE.0) GO TO 150
      NCPENT=NCPE(J)
      NOPENT=NOPE(J)
      STATJ=LFSTAT(J)
      IF (NCPENT.GT.1.OR.NOPENT.GT.1) WRITE (W,3010)
      IF (NCPENT.GT.1.OR.NOPENT.GT.1) LINES=LINES+1
      NEND=MAX0(NCPENT,NOPENT)
3025  FORMAT (1X,I4,3X,A8,2X,A8,1X,2F12.0,F14.0,3X,I4,F12.0,3F13.0)
      DO 140 N=1,NEND
      NM=BL
      STA=BL
      ALLT=0.0
      AULT=0.0
      THRUX=0.0
      ENDPX=0
      AMTPX=0.
      COSTCX=0.0
      THRUOX=0.
      COSTOX=0.
      IF (N.GT.1) GO TO 120
      NM=LFLNAM(J)
      STA=STATLB(STATJ+1)
      ALLT=LFLLL(J)
      AULT=LFLUL(J)
 120  IF (N.GT.NCPENT) GO TO 130
      THRUX=THRU(J,N)
      ENDPX=ENDPAY(J,N)
      AMTPX=AMTPAY(J,N)
      COSTCX=COSTC(J,N)
 130  IF (N.GT.NOPENT) GO TO 180
      THRUOX=THRUO(J,N)
      COSTOX=COSTO(J,N)
 180  WRITE (W,3025) LFLION(J),NM,STA,ALLT,AULT,THRUX,ENDPX,AMTPX,
     1 COSTCX,THRUOX,COSTOX
      LINES=LINES+1
      IF (LINES.LT.LINPAG) GO TO 140
      CALL HEADER(5)
      WRITE (W,3000)
      LINES=16
 140  CONTINUE
      IF (NCPENT.GT.1.OR.NOPENT.GT.1) WRITE (W,3010)
      IF (NCPENT.GT.1.OR.NOPENT.GT.1) LINES=LINES+1
 150  CONTINUE
 155  IF (IWAY0.EQ.0) CALL HEADER(5)
      IF (IWAY0.EQ.0) LINES=10
      IF (IWAY1.EQ.0) GO TO 235
C
C  PRINT THE RESULTS FOR LANDFILLS WITH LUMP PAYMENTS
C  FOR CAPITAL COSTS
C
 3034 FORMAT (///1X,"LANDFILLS WITH LUMP PAYMENTS OF CAPITAL COST"/)
 3035 FORMAT (/T47,"ANNUAL",T104,"ANNUAL",T114,"ANNUAL",T126,
     1 "ANNUAL"/T26,"THROUGHPUT,TONNES",T46,"CAPACITY",T62,
     2 "LUMP PAYMENTS OF CAPITAL COST",T103,"CAPITAL",T112,
     3 "THROUGHPUT",T124,"OPERATING"/1X,"LABEL",T9,"NAME",T17,
     4 "STATUS",T28,"LEAST",T38,"MOST",T47,"TONNES",T56,
     5 3("YEAR   -    $     "),T104,"COST,$",T114,"TONNES",T126,
     6 "COST,$"/)
 3045 FORMAT (1X,I4,1X,A8,1X,A8,2F10.0,F11.0,3(1X,I4,F10.0),3F11.0)
      IF (LINES.LT.LINPAG-12) GO TO 215
      CALL HEADER(5)
      LINES=10
  215 WRITE (W,3034)
  220 WRITE (W,3035)
      LINES=LINES+8
      DO 230 J=1,NUMLFL
      IF (IWAY(J).NE.1) GO TO 230
      NCPENT=NCPE(J)
      NOPENT=NCPE(J)
      STATJ=LFSTAT(J)
      IF (NCPENT.GT.1.OR.NOPENT.GT.1) WRITE (W,3010)
      IF (NCPENT.GT.1.OR.NOPENT.GT.1) LINES=LINES+1
      NEND=MAX0(NCPENT,NOPENT)
      DO 240 N=1,NEND
      NM=BL
      STA=BL
      ALLT=0.
      AULT=0.
      THRUX=0.
      NYE(1)=0
      NYE(2)=0
      NYE(3)=0
      ALU(1)=0.
      ALU(2)=0.
      ALU(3)=0.
      COSTCX=0.
      THRUOX=0.
      COSTOX=0.
      IF (N.GT.1) GO TO 250
      NM=LFLNAM(J)
      STA=STATEB(STATJ+1)
      ALLT=LFLLL(J)
      AULT=LFLUL(J)
 250  IF (N.GT.NCPENT) GO TO 265
      THRUX=THRU(J,N)
      DO 260 I=1,3
      NYE(I)=IYR(J,N,I)
 260  ALU(I)=LMPPAY(J,N,I)
 265  COSTCX=COSTC(J,N)
      IF (N.GT.NOPENT) GO TO 270
      THRUOX=THRU(J,N)
      COSTOX=COSTO(J,N)
 270  WRITE (W,3045) LFLIDN(J),NM,STA,ALLT,AULT,THRUX,(NYE(I),ALU(I),
     1 I=1,3),COSTCX,THRUOX,COSTOX
      LINES=LINES+1
      IF (LINES.LT.LINPAG) GO TO 240
      CALL HEADER(5)
      WRITE (W,3034)
      WRITE (W,3035)
      LINES=16
 240  CONTINUE
      IF (NCPENT.GT.1.OR.NOPENT.GT.1) WRITE (W,3010)
      IF (NCPENT.GT.1.OR.NOPENT.GT.1) LINES=LINES+1
 230  CONTINUE
 235  IF (INCP.EQ.0.AND.INOP.EQ.0) RETURN
      IF (LINES.LT.LINPAG-9) GO TO 325
      CALL HEADER(5)
      LINES=10
  325 WRITE (W,3050)
 3050 FORMAT (//1X,"LANDFILLS - RESULTS OF REGRESSION"//1X,
     1 "LABEL",T10,"NAME",T22,"INTERCEPT",T37,"SLOPE",T48,"RSQUARE",
     2 T58,"CAPITAL COST VS. CAPACITY",3X,"OPERATIONAL COST VS.THROUGHPU
     3T"/)
      LINES=LINES+5
      DO 310 J=1,NUMLFL
      IF (LINES.LT.LINPAG) GO TO 330
      CALL HEADER(5)
      WRITE (W,3050)
 330  NOPENT=NOPE(J)
      NCPENT=NCPE(J)
      IF (NOPENT.EQ.1.AND.NCPENT.EQ.1) GO TO 310
      IF (NOPENT.EQ.1) GO TO 320
      WRITE (W,3060) LFLIDN(J),LFLNAM(J),FC(J,2),VC(J,2),RS(J,2)
      LINES=LINES+1
 3060 FORMAT (1X,I4,2X,A8,F15.1,F13.5,F10.3,T68,"X")
  320 IF (NCPENT.EQ.1) GO TO 310
      WRITE (W,3070) LFLIDN(J),LFLNAM(J),FC(J,1),VC(J,1),RS(J,1)
 3070 FORMAT (1X,I4,2X,A8,F15.1,F13.5,F10.3,T96,"X")

      LINES=LINES+1
  310 CONTINUE
      RETURN
      END
      SUBROUTINE MARKET
C
C        ***************************************************************
C        *                                                             *
C        *    THIS ROUTINE READS IN RESOURCE NAMES, THEN DEMAND        *
C        *                                                             *
C        ***************************************************************
C
C     ***    INITIAL DECLARATION OF L VARIABLES AS REAL
C
      IMPLICIT REAL(L)
C
C     ***    DECLARATION OF VARIABLES FOR NAMES AND
C            TITLES
      REAL ASTRSK
      REAL TITLE,RUNNAM,VECNAM,SRCNAM,TRFNAM,PLTNAM,
     1        LFLNAM,RSCNAM,MKTNAM
C
      INTEGER R,W,T,D
      INTEGER TOTERR,ERFLAG,RUNYRS,
     1           RUNBEG,RUNEND,INREP,OUTREP,TWRITE,DWRITE,LANDCO,
     2            SRCIDN,TRFIDN,PLTIDN,LFLIDN,MKTIDN,
     3           ORIGID,DESTID,TRSTAT,PLSTAT,LFSTAT,STATJ
C
C     *** GENERAL COMMON BLOCK
C
      COMMON/GENDAT/ASTRSK(15),TITLE(5),RUNNAM(5),
     1              LTOLFL,LNUMTR,LNUMPL,LNUMLF,LCAPTL,RATE,
     2              R,W,T,D,TOTERR,ERFLAG,DAT,
     3              RUNBEG,RUNEND,RUNYRS,INREP,OUTREP,TWRITE,DWRITE,
     4              LANDCD,NUMVEC,NUMSRC,NUMTRF,NUMPLT,NUMLFL,NRESRC,
     5              NUMMKT(9),NPAGE,LINPAG
C
C     ***    COMMON FOR MARKETS
      COMMON/MKTDAT/RSCNAM(9),MKTNAM(9,9),DEMAND(9,9),PRICET(9,9),
     1                                                  MKTIDN(9,9)
C
  1   CONTINUE
      LINES=16
      IF(INREP.GT.0) CALL HEADER(6)
      IF(INREP.GT.0) WRITE(W,5000)
 5000  FORMAT(///T53,"ANNUAL",T65,"TOTAL"/T4,"RESOURCE",T19,
    1 "MARKET",T27,"MARKET",T40,"PRICE",T53,"DEMAND",
    2 T65,"DEMAND"/1X,"NUMBER    NAME",T19,"LABEL",
    3 T28."NAME",T39,"$/TONNE",T53,"TONNES",T65,"TONNES"/)
 5040 FORMAT (1X)
        FACTOR = USPWF(RATE,RUNYRS)/RUNYRS
 5010 FORMAT(A8)
     DO 720 J=1, NRESRC
     READ(R,5010)RSCNAM(J)
     IF (INREP.EQ.0.OR.LINES.LE.LINPAG) GO TO 10
     CALL HEADER(6)
     WRITE (W,5000)
     LINES=16
   10 NMKTRJ=NUMMKT(J)
     DO 710 K=1,NMKTRJ
     READ(R,5020)MKTIDN(J,K),MKTNAM(J,K),PRICE,DEMNND
 5020 FORMAT(I3,A8,9X,F10.3,F10.0)
      PRICET(J,K) = -PRICE *FACTOR
      DEMAND(J,K) = DEMNND *RUNYRS
      IF (INREP.EQ.0) GO TO 701
      WRITE(W,5030) J,RSCNAM(J),MKTION(J,K),MKTNAM(J,K),PRICE,DEMNND,
     1DEMAND(J,K)
 5030 FORMAT(1X,I3,5X,A8,1X,I4,4X,A8,F11.2,2X,2F12.0)
      LINES=LINES+1
      IF (LINES.LE.LINPAG) GO TO 701
      CALL HEADER(6)
      WRITE (W,5000)
      LINES=16
C
 701    JCODE=MKTION(J,K)/100
      IF(JCODE.NE.J)  CALL ERROR(601)
 710    CONTINUE
      IF (INREP.GT.0) WRITE (W,5040)
      LINES=LINES+1
 720  CONTINUE
      RETURN
      END
```

```
C
      SUBROUTINE LINKEM
C
C       ****************************************************************
C       *                                                              *
C       *   THIS ROUTINE READS IN ALL TRANSPORTATION LINKS,            *
C       *   CHECKS FOR OBVIOUS DATA ERRORS AND RECORDS INFORMATION     *
C       *   IN ARRAY ORIGID( ),DESTID( ) AND ARCOST( ).                *
C       *   ACCORDING TO THE FLAG INREP, ARCS MAY BE LISTED            *
C       ****************************************************************
C
C     ***   INITIAL DECLARATION OF L VARIABLES AS REAL
C
      IMPLICIT REAL(L)
C
C     ***   DECLARATION OF DOUBLE PRECISION VARIABLES FOR NAMES
C           TITLES
      REAL ASTRSK
      REAL TITLE,RUNNAM,VECNAM,SRCNAM,TRFNAM,PLTNAM,
     1       LFLNAM,RSCNAM,MKTNAM
      REAL NAME1,NAME2,QUESTS
C
      INTEGER R,W,T,D
      INTEGER TOTERR,ERFLAG,RUNYRS,
     1          RUNBEG,RUNEND,INREP,OUTREP,TWRITE,DWRITE,LANDCD,
     2           SRCION,TRFION,PLTION,LFLION,MKTION,
     3          ORIGID,DESTID,TRSTAT,PLSTAT,LFSTAT,STATJ
C
C     *** GENERAL COMMON BLOCK
C
      COMMON/GENDAT/ASTRSK(15),TITLE(5),RUNNAM(5),
     1              LTOLFL,LNUMTR,LNUMPL,LNUMLF,LCAPTL,RATE,
     2              R,W,T,D,TOTERR,ERFLAG,DAT,
     3              RUNBEG,RUNEND,RUNYRS,INREP,OUTREP,TWRITE,DWRITE,
     4              LANDCD,NUMVEC,NUMSRC,NUMTRF,NUMPLT,NUMLFL,NRESRC,
     5              NUMMKT(9),NPAGE,LINPAG
C
C     ***   VEHICLES COMMON BLOCK
C
      COMMON/VECDAT/VECNAM(8),COSTTH(8)
C
C     ***   SOURCES COMMON BLOCK
C
      COMMON/SRCDAT/SRCNAM(200),SRCTOT(200),ALWAST,SRCION(200)
C
C     ***   TRANSFER STATIONS COMMON BLOCK
C
      COMMON/TRFDAT/TRFNAM(20),TRFLLT(20),TRFULT(20),
     1              TRFFCP(20),TRFVCP(20),TRFFOP(20),TRFVOP(20),
     2              TRFION(20),TRSTAT(20)
C
C     ***   COMMON FOR PLANTS
C
      COMMON/PLTDAT/PLTNAM(20),PLTLLT(20),PLTULT(20),
     1              PLTFCP(20),PLTVCP(20),PLTFOP(20),PLTVOP(20),
     2              PLTKEP(20),PLTRKN(20,9),
     3              PLTION(20),PLSTAT(20)
```

```
C
C     ***   COMMON FOR LANDFILLS
C
      COMMON/LFLDAT/LFLNAM(20),LFLLLT(20),LFLULT(20),
    1               LFLFCP(20),LFLVCP(20),LFLFOP(20),LFLVOP(20),
    2               LFLIDN(20),LFSTAT(20)
C
C     ***   COMMON FOR MARKETS
      COMMON/MKTDAT/RSCNAM(9),MKTNAM(9,9),DEMAND(9,9),PRICET(9,9),
    1                                                  MKTIDN(9,9)
C
C     ***   COMMON FOR TRANSPORTATION ARCS
C
      COMMON/LNKDAT/ARCOST(5000),ORIGID(5000),DESTID(5000),
    1               NANDST,NANDSP,NANDSL,NANDTP,NANDTL,NANDPL,NANDPM
C
C
      REAL ATYPE(5)
      DATA ATYPE /" SR "," LF "," TR "," PL "," MR "/
      DATA QUESTS/"&&&&&&&&"/
   1  CONTINUE
      LINES=10
      IF(INREP.GT.1) CALL HEADER(7)
      IF(INREP.EQ.2) WRITE(W,8000)
 8000 FORMAT(///T69,"ROUND",T78,"COST, $"/11X,7("-"),"ORIGIN",
    1 7("-"),T36,"-----DESTINATION-----",T58,"VEHICLE",T70,"TRIP",
    2 T78,"PER TONNE"/
    3 11X,"LABEL    NAME",T28,"TYPE",T36,"LABEL    NAME",T52,
    4 "TYPE",T60,"TYPE",T70,"MIN",T78,"PER TRIP"/)
     FACTOR=USPWF(RATE,RUNYRS)/RUNYRS
C
 8010 FORMAT(I1,1X,I3,2X,I3,4X,I1,F5.0)
 8020 FORMAT(11X,I4,2X,A8,2X,A4,4X,I4,2X,A8,2X,A4,2X,A8,F8.1,F12.5)
C
C    ***    READ IN SOURCE TO TRANSFER STATION ARCS
     NANDST=0
     DO 110 J=1,4999
     ERFLAG=0
     READ(R,8010)ICODE,IDORIG,IDDEST,IDVEIC,TIME
     IF(ICODE.EQ.9) GO TO 200
     IF(ICODE.NE.1) CALL ERROR (701)
     IF(IDORIG.LT.100) CALL ERROR(711)
     IF(IDDEST.LT.100) CALL ERROR(712)
     IF(IDVEIC.LE.0) CALL ERROR(721)
     IF(IDVEIC.GT.8) CALL ERROR(721)
     IF(ERFLAG.EQ.1) GO TO 110
     NANDST=NANDST+1
     ORIGID(NANDST)=IDORIG
     DESTID(NANDST)=IDDEST
     COST=TIME* COSTTH(IDVEIC)/60.
     ARCOST(NANDST)= COST *FACTOR
C
     IF(INREP.NE.2) GO TO 110
C
      NAME1=QUESTS
C
     DO 101 K=1,NUMSRC
```

```
101     IF(SRCIDN(K).EQ.IDORIG)NAME1=SRCNAM(K)
         NAME2=QUESTS
       DO 102 K=1,NUMTRF
102    IF(TRFIDN(K).EQ.IDDEST)NAME2 =TRFNAM(K)
        IF(NAME1.EQ.QUESTS) CALL ERROR(731)
        IF(NAME2.EQ.QUESTS) CALL ERROR(732)
       WRITE(W,8020) IDORIG,NAME1,ATYPE(1),IDDEST,NAME2,ATYPE(3),
     1    VECNAM(IDVEIC),TIME,ARCOST(NANDST)
       LINES=LINES+1
       IF (LINES.LT.LINPAG) GO TO 110
       CALL HEADER(7)
       WRITE (W,8000)
       LINES=10
110    CONTINUE
C      ***     READ IN SOURCE TO PLANT ARCS
200    NANDSP=NANDST
       IF (INREP.EQ.2) WRITE (W,8030)
       LINES=LINES+1
8030   FORMAT (1X)
       DO 210 J=1,4999
       ERFLAG=0
       READ(R,8010)ICODE,IDORIG,IDDEST,IDVEIC,TIME
       IF(ICODE.EQ.9)GO TO 300
       IF(ICODE.NE.2) CALL ERROR(702)
       IF(IDORIG.LT.100) CALL ERROR(711)
       IF(IDDEST.LT.100) CALL ERROR(712)
       IF(IDVEIC.LE.0) CALL ERROR(721)
       IF(IDVEIC.GT.8) CALL ERROR(721)
       NANDSP=NANDSP+1
       ORIGID(NANDSP) = IDORIG
       DESTID(NANDSP) = IDDEST
       COST =TIME * COSTTH(IDVEIC)/60.
       ARCOST(NANDSP) = COST *FACTOR
       IF(INREP.NE.2) GO TO 210
        NAME1=QUESTS
       DO 201 K=1,NUMSRC
201    IF(SRCIDN(K).EQ.IDORIG)NAME1 = SRCNAM(K)
        NAME2=QUESTS
       DO 202 K=1,NUMPLT
202    IF(PLTIDN(K).EQ.IDDEST)NAME2 = PLTNAM(K)
        IF(NAME1.EQ.QUESTS) CALL ERROR(731)
        IF(NAME2.EQ.QUESTS) CALL ERROR(733)
       WRITE(W,8020) IDORIG,NAME1,ATYPE(1),IDDEST,NAME2,ATYPE(4),
     1    VECNAM(IDVEIC),TIME,ARCOST(NANDSP)
       LINES=LINES+1
       IF (LINES.LT.LINPAG) GO TO 210
       CALL HEADER(7)
       WRITE (W,8000)
       LINES=10
210    CONTINUE
C
C      ***     READ IN SOURCE TO LANDFILL ARCS
300    NANDSL = NANDSP
       IF (INREP.EQ.2) WRITE (W,8030)
       LINES=LINES+1
       DO 310 J=1,4999
       ERFLAG=0
```

```
      READ(R,8010)ICODE,IDORIG,IDDEST,IDVEIC,TIME
      IF(ICODE.EQ.9) GO TO 400
      IF(ICODE.NE.3) CALL ERROR(703)
      IF(IDORIG.LT.100) CALL ERROR(711)
      IF(IDDEST.LT.100) CALL ERROR(712)
      IF(IDVEIC.LE.0) CALL ERROR(721)
      IF(IDVEIC.GT.8) CALL ERROR(721)
      IF(ERFLAG.EQ.1) GO TO 310
      NANDSL = NANDSL +1
      ORIGID(NANDSL) = IDORIG
      DESTID(NANDSL) = IDDEST
      COST = TIME * COSTTH(IDVEIC)/60.
      ARCOST(NANDSL) = COST * FACTOR
      IF(INREP.NE.2) GO TO 310
       NAME1=QUESTS
      DO 301 K=1,NUMSRC
  301 IF(SRCIDN(K).EQ.IDORIG) NAME1 = SRCNAM(K)
       NAME2=QUESTS
      DO 302 K=1,NUMLFL
  302 IF(LFLIDN(K).EQ.IDDEST) NAME2 = LFLNAM(K)
        IF(NAME1.EQ.QUESTS) CALL ERROR(731)
        IF(NAME2.EQ.QUESTS) CALL ERROR(734)
      WRITE(W,8020) IDORIG,NAME1,ATYPE(1),IDDEST,NAME2,ATYPE(2),
     1   VECNAM(IDVEIC),TIME,ARCOST(NANDSL)
      LINES=LINES+1
      IF (LINES.LT.LINPAG) GO TO 310
      CALL HEADER(7)
      WRITE (W,8000)
      LINES=10
  310 CONTINUE
C
C     ***     READ IN TRANSFER TO PLANT ARCS
C
  400 NANDTP=NANDSL
      IF (INREP.EQ.2) WRITE (W,8030)
      LINES=LINES+1
      DO 410 J=1,4999
      ERFLAG = 0
      READ(R,8010)ICODE,IDORIG,IDDEST,IDVEIC,TIME
      IF(ICODE.EQ.9) GO TO 500
      IF(ICODE.NE.4) CALL ERROR(704)
      IF(IDORIG.LT.100) CALL ERROR(711)
      IF(IDDEST.LT.100) CALL ERROR(712)
      IF(IDVEIC.LE.0)  CALL ERROR(721)
      IF(IDVEIC.GT.8)  CALL ERROR(721)
      IF(ERFLAG.EQ.1) GO TO 410
      NANDTP=NANDTP+1
      ORIGID(NANDTP) = IDORIG
      DESTID(NANDTP) = IDDEST
      COST = TIME * COSTTH(IDVEIC)/60.
      ARCOST(NANDTP) = COST * FACTOR
      IF(INREP.NE.2) GO TO 410
       NAME1=QUESTS
      DO 401 K=1,NUMTRF
  401 IF(TRFIDN(K).EQ.IDORIG)NAME1 = TRFNAM(K)
       NAME2=QUESTS   .
      DO 402 K=1,NUMPLT
  402 IF(PLTION(K).EQ.IDDEST) NAME2 = PLTNAM(K)
       IF(NAME1.EQ.QUESTS) CALL ERROR(732)
       IF(NAME2.EQ.QUESTS) CALL ERROR(733)
      WRITE(W,8020) IDORIG,NAME1,ATYPE(3),IDDEST,NAME2,ATYPE(4),
     1   VECNAM(IDVEIC),TIME,ARCOST(NANDTP)
      LINES=LINES+1
      IF (LINES.LT.LINPAG) GO TO 410
      CALL HEADER(7)
      WRITE (W,8000)
      LINES=10
  410 CONTINUE
C
C     ***    READ IN TRANSFER TO LANDFILL ARCS
C
  500 NANDTL = NANDTP
      IF (INREP.EQ.2) WRITE (W,8030)
      LINES=LINES+1
      DO 510 J=1,4999
      ERFLAG = 0
      READ(R,8010) ICODE,IDORIG,IDDEST,IDVEIC,TIME
      IF(ICODE.EQ.9) GO TO 600
      IF(ICODE.NE.5) CALL ERROR(705)
      IF(IDORIG.LT.100) CALL ERROR(711)
      IF(IDDEST.LT.100) CALL ERROR(712)
      IF(IDVEIC.LE.0)  CALL ERROR(721)
      IF(IDVEIC.GT.8)  CALL ERROR(721)
      IF(ERFLAG.EQ.1) GO TO 510
      NANDTL = NANDTL+1
      ORIGID(NANDTL) = IDORIG
      DESTID(NANDTL) = IDDEST
      COST = TIME * COSTTH(IDVEIC)/60.
      ARCOST(NANDTL) = COST * FACTOR
      IF(INREP.NE.2) GO TO 510
      NAME1=QUESTS
      DO 501 K=1,NUMTRF
  501 IF(TRFION(K).EQ.IDORIG)NAME1 = TRFNAM(K)
      NAME2=QUESTS
      DO 502 K=1,NUMLFL
  502 IF(LFLION(K).EQ.IDDEST)NAME2 = LFLNAM(K)
       IF(NAME1.EQ.QUESTS) CALL ERROR(732)
       IF(NAME2.EQ.QUESTS) CALL ERROR(734)
      WRITE(W,8020) IDORIG,NAME1,ATYPE(3),IDDEST,NAME2,ATYPE(2),
     1   VECNAM(IDVEIC),TIME,ARCOST(NANDTL)
      LINES=LINES+1
      IF (LINES.LT.LINPAG) GO TO 510
      CALL HEADER(7)
      WRITE (W,8000)
      LINES=10
  510 CONTINUE
C
C     ***    READ IN PLANT TO LANDFILL ARCS
C
  600 NANDPL=NANDTL
      IF (INREP.EQ.2) WRITE (W,8030)
      LINES=LINES+1
      DO 610 J=1,4999
      ERFLAG=0
      READ (R,8010) ICODE,IDORIG,IDDEST,IDVEIC,TIME
      IF(ICODE.EQ.9) GO TO 700
      IF(ICODE.NE.6) CALL ERROR(706)
      IF(IDORIG.LT.100) CALL ERROR(711)
      IF(IDDEST.LT.100) CALL ERROR(712)
      IF(IDVEIC.LE.0) CALL ERROR(721)
      IF(IDVEIC.GT.8) CALL ERROR(721)
      IF(ERFLAG.EQ.1) GO TO 610
      NANDPL = NANDPL+1
      ORIGID(NANDPL) = IDORIG
      DESTID(NANDPL) = IDDEST
      COST = TIME *COSTTH(IDVEIC)/60.
      ARCOST(NANDPL) = COST * FACTOR
      IF(INREP.NE.2) GO TO 610
       NAME1=QUESTS
      DO 601 K=1,NUMPLT
  601 IF(PLTION(K).EQ.IDORIG)NAME1 = PLTNAM(K)
       NAME2=QUESTS
      DO 602 K=1,NUMLFL
  602 IF(LFLION(K).EQ.IDDEST)NAME2 = LFLNAM(K)
       IF(NAME1.EQ.QUESTS) CALL ERROR(733)
       IF(NAME2.EQ.QUESTS) CALL ERROR(734)
      WRITE(W,6020) IDORIG,NAME1,ATYPE(4),IDDEST,NAME2,ATYPE(2),
     1   VECNAM(IDVEIC),TIME,ARCOST(NANDPL)
      LINES=LINES+1
      IF (LINES.LT.LINPAG) GO TO 610
      CALL HEADER(7)
      WRITE (W,8000)
      LINES=10
  610 CONTINUE
C
C     ***     READ IN PLANT TO MARKET ARCS
C
  700 NANDPM=NANDPL
      IF (INREP.EQ.2) WRITE (W,8030)
      LINES=LINES+1
      DO 710 J=1,4999
      ERFLAG=0
      READ(R,8010) ICODE,IDORIG,IDDEST,IDVEIC,TIME
      IF(ICODE.EQ.9) GO TO 800
      IF(ICODE.NE.7) CALL ERROR(707)
      IF(IDORIG.LT.100) CALL ERROR(711)
      IF(IDDEST.LT.100) CALL ERROR(712)
      IF(IDVEIC.LE.0) CALL ERROR(721)
      IF(IDVEIC.GT.8) CALL ERROR(721)
      IF(ERFLAG.EQ.1) GO TO 710
      NANDPM=NANDPM+1
      ORIGID(NANDPM)=IDORIG
      DESTID(NANDPM)=IDDEST
      COST=TIME*COSTTH(IDVEIC)/60.
      ARCOST(NANDPM)=COST*FACTOR
       IF(INREP.NE.2) GO TO 710
       NAME1=QUESTS
       DO 701 K=1,NUMPLT
  701  IF(PLTION(K).EQ.IDORIG) NAME1=PLTNAM(K)
       NAME2=QUESTS
       DO 702 I=1,NRESKC
       NMKTRI=NUMMKT(I)                                                  00200700
       DO 702 K=1,NMKTRI                                                 00200800
702    IF (MKTIDN(I,K).EQ.IDDEST) NAME2=MKTNAM(I,K)                      00200900
       IF (NAME1.EQ.QUESTS) CALL ERROR(733)                              00201000
       IF (NAME2.EQ.QUESTS) CALL ERROR(735)                              00201100
      WRITE(W,8020) IDORIG,NAME1,ATYPE(4),IDDEST,NAME2,ATYPE(5),         00201200
     1   VECNAM(IDVEIC),TIME,ARCOST(NANDPM)                              00201300
      LINES=LINES+1                                                      00201400
      IF (LINES.LT.LINPAG) GO TO 710                                     00201500
      CALL HEADER(7)                                                     00201600
      WRITE (W,8000)                                                     00201700
      LINES=10                                                           00201800
710   CONTINUE                                                           00201900
800   IF(INREP.EQ.2) WRITE(W,8050)                                       00202000
8050  FORMAT(//20X,"END OF TRANSPORTATION ROUTES.")                      00202100
      RETURN                                                             00202200
      END                                                                00202300
      SUBROUTINE TSETUP
C
C       ****************************************************************
C       *                                                              *
C       *  THIS ROUTINE SETS UP ON DEVICE T THE FILE FOR INPUT TO      *
C       *  THE MILP SOFTWARE PACKAGE. ALL VARIABLES EXCEPT             *
C       *  ONE AND UNN  ARE PASSED FROM EARLIER ROUTINES.              *
C       *                                                              *
C       ****************************************************************
C
C     ***   INITIAL DECLARATION OF L VARIABLES AS REAL
C
      IMPLICIT REAL(L)
C
C     ***   DECLARATION OF VARIABLES FOR NAMES AND
C           TITLES
      REAL ASTRSK
      REAL TITLE,RUNNAM,VECNAM,SRCNAM,TRFNAM,PLTNAM,
     1        LFLNAM,RSCNAM,MKTNAM
C
      INTEGER R,W,T,D
      INTEGER TOTERR,ERFLAG,RUNYRS,
     1             RUNBEG,RUNEND,INREP,OUTREP,TWRITE,DWRITE,LANDCD,
     2              SRCIDN,TRFIDN,PLTIDN,LFLIDN,MKTIDN,
     3             ORIGID,DESTID,TRSTAT,PLSTAT,LFSTAT,STATJ
C
C     *** GENERAL COMMON BLOCK
C
      COMMON/GENDAT/ASTRSK(15),TITLE(5),RUNNAM(5),
     1                  LTOLFL,LNUMTR,LNUMPL,LNUMLF,LCAPTL,RATE,
     2                  R,W,T,D,TOTERR,ERFLAG,DAT,
     3                  RUNBEG,RUNEND,RUNYRS,INREP,OUTREP,TWRITE,DWRITE,
     4                  LANDCD,NUMVEC,NUMSRC,NUMTRF,NUMPLT,NUMLFL,NRESRC,
     5                  NUMMKT(9),NPAGE,LINPAG
C
C     ***   VEHICLES COMMON BLOCK
C
      COMMON/VECDAT/VECNAM(8),COSTTH(8)
C
C     ***   SOURCES COMMON BLOCK
C
      COMMON/SRCDAT/SRCNAM(200),SRCTOT(200),ALWAST,SRCIDN(200)
C
C     ***   TRANSFER STATIONS COMMON BLOCK
C
      COMMON/TRFDAT/TRFNAM(20),TRFLLT(20),TRFULT(20),
     1                  TRFFCP(20),TRFVCP(20),TRFFOP(20),TRFVOP(20),
     2                  TRFIDN(20),TRSTAT(20)
C
C     ***   COMMON FOR PLANTS
C
      COMMON/PLTDAT/PLTNAM(20),PLTLLT(20),PLTULT(20),
     1                  PLTFCP(20),PLTVCP(20),PLTFOP(20),PLTVOP(20),
     2                  PLTKEP(20),PLTRRN(20,9),
     3                  PLTIDN(20),PLSTAT(20)
C
C     ***   COMMON FOR LANDFILLS
```

```
C
      COMMON/LFLDAT/LFLNAM(20),LFLLLT(20),LFLULT(20),
     1                LFLFCP(20),LFLVCP(20),LFLFUP(20),LFLVGP(20),
     2                LFLION(20),LFSTAT(20)
C
C     ***   COMMON FOR MARKETS
      COMMON/MKTDAT/RSCNAM(9),MKTNAM(9,9),DEMAND(9,9),PRICET(9,9),
     1                                                  MKTION(9,9)
C
C     ***   COMMON FOR TRANSPORTATION ARCS
C
      COMMON/LNKDAT/ARCOST(5000),ORIGID(5000),DESTID(5000),
     1                NANDST,NANDSP,NANDSL,NANDTP,NANDTL,NANDPL,NANDPM
C
C     ***  MANY OF THE CHARACTER STRINGS, E.G. ROW NAMES, REPEAT
C     ***   SO ARE DEFINED INITIALLY AS ARRAY ELEMENTS
C
      REAL SRCROW,TRFROW(4),PLTROW(5),LFLROW(3),
     1                  MKTROW(2),LMTROW(4),CSTROW(5),CHGROW(4),
     2                  RHSNAM(5),BNDNAM
      DIMENSION RWTYPE(4),ARCTYP(8),WQTYTO(4),
     1            DELTAS(4),FXUP(4),FXUPV(4)
      DATA RWTYPE/" E"," L"," G"," N"/
      DATA SRCROW/"SRCGN"/,
     1       TRFROW/"TRFTO","TRLES","TRMOS","TRFRM"/,
     2       PLTROW/"PLNTO","PLEST","PLMOS","PLFRM","PLAR"/,
     3       LFLROW/"LFLTO","LFLES","LFMOS"/,
     4       MKTROW/"MAKTO","MKMOS"/,
     5       LMTROW/"LFLTOTAL","NUMTRANF","NUMPLANT","NUMLNDFL"/,
     6       CSTROW/"CAPITALC","OPERATNG","TRANSPRT","REVENUES",
     7                "NETCOST "/,
     8       CHGROW/"CHCAPITL","CHOPRTNG","CHTRNPRT","CHREVENU"/
C
      DATA ARCTYP/"ST","SP","SL","TP","TL","PL","PM","  "/
C
      DATA WQTYTO/"WTRF","WPLT","WLFL","WMKT"/
C
      DATA DELTAS/"DTRF","DPLT","DLFL","    "/
C
      DATA RHSNAM/"ORIGRHS ","CHGGEN  ","CHGDMD  ","CHGTLFL ",
     1                "CHGCPL  "/
      DATA BNDNAM/"FACILITY"/
      DATA FXUP/"FX","FX","UP","  "/
      DATA FXUPV/0.,1.,1.,0./
C
      IF(INREP.GT.0) CALL HEADER(3)
      ONE=1.
      ONN=-1.
C     ***   WRITE DATA-SET NAME
C
      WRITE(T,9110)
 9110 FORMAT("NAME",10X,"UNTWAST ")
C
C     ***   WRITE ROWS SECTION   ****************************
C
      WRITE(T,9120)
 9120 FORMAT("ROWS")
```

```
       DO 110 J=1,NUMSRC
 110   WRITE(T,9130)RWTYPE(1),SRCROW,SRCIDN(J)
 9130  FORMAT(1X,A2,1X,A5,I3)
C
C
C
C               BYPASS IF NO TRANSFER STATIONS
C
      IF(NUMTRF.EQ.0) GO TO 125
       DO 120 J=1,NUMTRF
       WRITE(T,9130)RWTYPE(1),TRFROW(1),TRFIDN(J)
       WRITE(T,9130)RWTYPE(2),TRFROW(2),TRFIDN(J)
       WRITE(T,9130)RWTYPE(3),TRFROW(3),TRFIDN(J)
       WRITE(T,9130)RWTYPE(1),TRFROW(4),TRFIDN(J)
 120   CONTINUE
C
C
C               BYPASS IF NO PLANTS
C
 125  IF(NUMPLT.EQ.0) GO TO 135
       DO 130 J=1,NUMPLT
       WRITE(T,9130)RWTYPE(1),PLTROW(1),PLTIDN(J)
       WRITE(T,9130)RWTYPE(2),PLTROW(2),PLTIDN(J)
       WRITE(T,9130)RWTYPE(3),PLTROW(3),PLTIDN(J)
       WRITE(T,9130)RWTYPE(1),PLTROW(4),PLTIDN(J)
       IF(NRESRC.EQ.0) GO TO 130
       DO 127 K=1,NRESRC
 127   WRITE(T,9140)RWTYPE(2),PLTROW(5),PLTIDN(J),K
 130   CONTINUE
 9140  FORMAT(1X,A2,1X,A4,I3,I1)
C
 135   DO 140 J=1,NUMLFL
       WRITE(T,9130)RWTYPE(1),LFLROW(1),LFLIDN(J)
       WRITE(T,9130)RWTYPE(2),LFLROW(2),LFLIDN(J)
       WRITE(T,9130)RWTYPE(3),LFLROW(3),LFLIDN(J)
 140   CONTINUE
C
C
C               BYPASS IF NO RESOURCES
C
      IF(NRESRC.EQ.0) GO TO 160
       DO 150 J=1,NRESRC
       NMKT=NUMMKT(J)
       DO 150 K=1,NMKT
       WRITE(T,9150)RWTYPE(1),MKTROW(1),MKTIDN(J,K)
       WRITE(T,9150)RWTYPE(2),MKTROW(2),MKTIDN(J,K)
 150   CONTINUE
 9150  FORMAT(1X,A2,1X,A5,I3)
C
 9170  FORMAT(1X,A2,1X,A6)
 160   WRITE(T,9170)RWTYPE(2),LMTROW(1)
       WRITE(T,9170)RWTYPE(2),LMTROW(2)
       WRITE(T,9170)RWTYPE(2),LMTROW(3)
       WRITE(T,9170)RWTYPE(2),LMTROW(4)
       WRITE(T,9170)RWTYPE(2),CSTROW(1)
       WRITE(T,9170)RWTYPE(4),CSTROW(2)
       WRITE(T,9170)RWTYPE(4),CSTROW(3)
       WRITE(T,9170)RWTYPE(4),CSTROW(4)
       WRITE(T,9170)RWTYPE(4),CSTROW(5)
C
       WRITE(T,9170)RWTYPE(4),CHGROW(1)
       WRITE(T,9170)RWTYPE(4),CHGROW(2)
       WRITE(T,9170)RWTYPE(4),CHGROW(3)
       WRITE(T,9170)RWTYPE(4),CHGROW(4)
C
C      ***   WRITE COLUMNS SECTION   ***************************
C
       WRITE(T,9210)
 9210  FORMAT("COLUMNS")
C
 9220  FORMAT(4X,A2,I3,I3,2X,A5,I3,2X,F12.6)
 9230  FORMAT(4X,A2,I3,I3,2X,A8,2X,F12.6)
C
C
C              BYPASS IF NO SOURCE TO TRANSFER ARCS
C
      IF(NANDST.EQ.0) GO TO 215
       DO 210 J=1,NANDST
      WRITE(T,9220)ARCTYP(1),ORIGID(J),DESTID(J),SRCROW,ORIGID(J),ONE
      WRITE(T,9220)ARCTYP(1),ORIGID(J),DESTID(J),TRFROW(1),DESTID(J),ONE
      WRITE(T,9230)ARCTYP(1),ORIGID(J),DESTID(J),CSTROW(3),ARCOST(J)
      WRITE(T,9230)ARCTYP(1),ORIGID(J),DESTID(J),CSTROW(5),ARCOST(J)
      WRITE(T,9230)ARCTYP(1),ORIGID(J),DESTID(J),CHGROW(3),ARCOST(J)
 210  CONTINUE
C
C
C              BYPASS IF NO SOURCE TO PLANT ARCS
C
 215  IF(NANDSP.EQ.NANDST) GO TO 225
      NNDST1=NANDST+1
      DO 220 J=NNDST1,NANDSP
      WRITE(T,9220)ARCTYP(2),ORIGID(J),DESTID(J),SRCROW,ORIGID(J),ONE
      WRITE(T,9220)ARCTYP(2),ORIGID(J),DESTID(J),PLTROW(1),DESTID(J),ONE
      WRITE(T,9230)ARCTYP(2),ORIGID(J),DESTID(J),CSTROW(3),ARCOST(J)
      WRITE(T,9230)ARCTYP(2),ORIGID(J),DESTID(J),CSTROW(5),ARCOST(J)
      WRITE(T,9230)ARCTYP(2),ORIGID(J),DESTID(J),CHGROW(3),ARCOST(J)
 220  CONTINUE
C
C
C              BYPASS IF NO SOURCE TO LANDFILL ARCS
C
 225  IF(NANDSL.EQ.NANDSP) GO TO 235
      NNDSP1=NANDSP+1
      DO 230 J=NNDSP1,NANDSL
      WRITE(T,9220)ARCTYP(3),ORIGID(J),DESTID(J),SRCROW,ORIGID(J),ONE
      WRITE(T,9220)ARCTYP(3),ORIGID(J),DESTID(J),LFLROW(1),DESTID(J),ONE
      WRITE(T,9230)ARCTYP(3),ORIGID(J),DESTID(J),CSTROW(3),ARCOST(J)
      WRITE(T,9230)ARCTYP(3),ORIGID(J),DESTID(J),CSTROW(5),ARCOST(J)
      WRITE(T,9230)ARCTYP(3),ORIGID(J),DESTID(J),CHGROW(3),ARCOST(J)
 230  CONTINUE
C
C
C              BYPASS IF NO TRANSFER TO PLANT ARCS
C
```

```
  235 IF(NANDTP.EQ.NANDSL) GO TO 245                                    00225200
      NNDSL1=NANDSL+1                                                   00225300
      DO 240 J=NNDSL1,NANDTP                                            00225400
      WRITE(T,9220)ARCTYP(4),ORIGID(J),DESTID(J),TRFROW(4),ORIGID(J),ONE00225500
      WRITE(T,9220)ARCTYP(4),ORIGID(J),DESTID(J),PLTROW(1),DESTID(J),ONE00225600
      WRITE(T,9230)ARCTYP(4),ORIGID(J),DESTID(J),CSTROW(3),ARCOST(J)    00225700
      WRITE(T,9230)ARCTYP(4),ORIGID(J),DESTID(J),CSTROW(5),ARCOST(J)    00225800
      WRITE(T,9230)ARCTYP(4),ORIGID(J),DESTID(J),CHGROW(3),ARCOST(J)    00225900
  240 CONTINUE                                                          00226000
C                                                                       00226100
C                                                                       00226200
C             BYPASS IF NO TRANSFER TO LANDFILL ARCS                    00226300
C                                                                       00226400
  245 IF(NANDTL.EQ.NANDTP) GO TO 255                                    00226500
      NNDTP1=NANDTP+1                                                   00226600
      DO 250 J=NNDTP1,NANDTL                                            00226700
      WRITE(T,9220)ARCTYP(5),ORIGID(J),DESTID(J),TRFROW(4),ORIGID(J),ONE00226800
      WRITE(T,9220)ARCTYP(5),ORIGID(J),DESTID(J),LFLROW(1),DESTID(J),ONE00226900
      WRITE(T,9230)ARCTYP(5),ORIGID(J),DESTID(J),CSTROW(3),ARCOST(J)    00227000
      WRITE(T,9230)ARCTYP(5),ORIGID(J),DESTID(J),CSTROW(5),ARCOST(J)    00227100
      WRITE(T,9230)ARCTYP(5),ORIGID(J),DESTID(J),CHGROW(3),ARCOST(J)    00227200
  250 CONTINUE                                                          00227300
C                                                                       00227400
C                                                                       00227500
C             BYPASS IF NO PLANT TO LANDFILL ARCS                       00227600
C                                                                       00227700
  255 IF(NANDPL.EQ.NANDTL) GO TO 265                                    00227800
      NNDTL1=NANDTL+1                                                   00227900
      DO 260 J=NNDTL1,NANDPL                                            00228000
      WRITE(T,9220)ARCTYP(6),ORIGID(J),DESTID(J),PLTROW(4),ORIGID(J),ONE00228100
      WRITE(T,9220)ARCTYP(6),ORIGID(J),DESTID(J),LFLROW(1),DESTID(J),ONE00228200
      WRITE(T,9230)ARCTYP(6),ORIGID(J),DESTID(J),CSTROW(3),ARCOST(J)    00228300
      WRITE(T,9230)ARCTYP(6),ORIGID(J),DESTID(J),CSTROW(5),ARCOST(J)    00228400
      WRITE(T,9230)ARCTYP(6),ORIGID(J),DESTID(J),CHGROW(3),ARCOST(J)    00228500
  260 CONTINUE                                                          00228600
C                                                                       00228700
C                                                                       00228800
C             BYPASS IF NO PLANT TO MARKET ARCS                         00228900
C                                                                       00229000
  265 IF(NANDPM.EQ.NANDPL) GO TO 300                                    00229100
      NNDPL1=NANDPL+1                                                   00229200
      DO 270 J=NNDPL1,NANDPM                                            00229300
      WRITE(T,9220)ARCTYP(7),ORIGID(J),DESTID(J),PLTROW(4),ORIGID(J),ONE00229400
       K=DESTID(J)/100                                                  00229500
      WRITE(T,9225)ARCTYP(7),ORIGID(J),DESTID(J),PLTROW(5),ORIGID(J),K, 00229600
     1                                                              ONE 00229700
      WRITE(T,9220)ARCTYP(7),ORIGID(J),DESTID(J),MKTROW(1),DESTID(J),ONE00229800
 9225  FORMAT(4X,A2,I3,I3,2X,A4,I3,I1,2X,F12.6)                         00229900
      WRITE(T,9230)ARCTYP(7),ORIGID(J),DESTID(J),CSTROW(3),ARCOST(J)    00230000
      WRITE(T,9230)ARCTYP(7),ORIGID(J),DESTID(J),CSTROW(5),ARCOST(J)    00230100
      WRITE(T,9230)ARCTYP(7),ORIGID(J),DESTID(J),CHGROW(3),ARCOST(J)    00230200
  270 CONTINUE                                                          00230300
C                                                                       00230400
C                                                                       00230500
C             BYPASS IF NO TRANSFER STATIONS                            00230600
C                                                                       00230700
 9320 FORMAT(4X,A4,I3,3X,A5,I3,2X,F12.6)                                00230800
 9330 FORMAT(4X,A4,I3,3X,A8,2X,F12.6)                                   00230900
 300  IF(NUMTRF.EQ.0) GO  TO 311                                        00231000
      DO 310 J=1,NUMTRF                                                 00231100
      WRITE(T,9320)HQTYTO(1),TRFION(J),TRFROW(1),TRFION(J),ONN          00231200
      WRITE(T,9320)HQTYTO(1),TRFION(J),TRFROW(2),TRFION(J),ONN          00231300
      WRITE(T,9320)HQTYTO(1),TRFION(J),TRFROW(3),TRFION(J),ONN          00231400
      WRITE(T,9320)HQTYTO(1),TRFION(J),TRFROW(4),TRFION(J),ONN          00231500
      WRITE(T,9330)HQTYTO(1),TRFION(J),CSTROW(1),TRFVCP(J)              00231600
      WRITE(T,9330)HQTYTO(1),TRFION(J),CSTROW(2),TRFVOP(J)              00231700
      TRFVAL=TRFVCP(J)+TRFVOP(J)                                        00231800
      WRITE(T,9330)HQTYTO(1),TRFION(J),CSTROW(5),TRFVAL                 00231900
 310  CONTINUE                                                          00232000
C                                                                       00232100
C                                                                       00232200
C              BYPASS IF NO PLANTS                                      00232300
C                                                                       00232400
 311  IF(NUMPLT.EQ.0) GO TO 335                                         00232500
      DO 320 J=1,NUMPLT                                                 00232600
      WRITE(T,9320)HQTYTO(2),PLTION(J),PLTROW(1),PLTION(J),ONN          00232700
      WRITE(T,9320)HQTYTO(2),PLTION(J),PLTROW(2),PLTION(J),ONN          00232800
      WRITE(T,9320)HQTYTO(2),PLTION(J),PLTROW(3),PLTION(J),ONN          00232900
      WRITE(T,9320)HQTYTO(2),PLTION(J),PLTROW(4),PLTION(J),PLTKEP(J)    00233000
C                                                                       00233100
C                                                                       00233200
C              BYPASS IF NO RESOURCE RECOVERY                           00233300
C                                                                       00233400
      IF(NRESRC.EQ.0) GO TO 316                                         00233500
      DO 315 K=1,NRESRC                                                 00233600
      WRITE(T,9340)HQTYTO(2),PLTION(J),PLTROW(5),PLTION(J),K,PLTRRN(J,K)00233700
 315  CONTINUE                                                          00233800
 9340 FORMAT(4X,A4,I3,3X,A4,I3,I1,2X,F12.6)                             00233900
 316  WRITE(T,9330)HQTYTO(2),PLTION(J),CSTROW(1),PLTVCP(J)              00234000
      WRITE(T,9330)HQTYTO(2),PLTION(J),CSTROW(2),PLTVOP(J)              00234100
      PLTVAL=PLTVCP(J)+PLTVOP(J)                                        00234200
      WRITE(T,9330)HQTYTO(2),PLTION(J),CSTROW(5),PLTVAL                 00234300
      WRITE(T,9330)HQTYTO(2),PLTION(J),CHGROW(1),PLTVCP(J)              00234400
      WRITE(T,9330)HQTYTO(2),PLTION(J),CHGROW(2),PLTVOP(J)              00234500
 320  CONTINUE                                                          00234600
      DO 330 J=1,NUMLFL                                                 00234700
      WRITE(T,9320)HQTYTO(3),LFLION(J),LFLROW(1),LFLION(J),ONN          00234800
      WRITE(T,9320)HQTYTO(3),LFLION(J),LFLROW(2),LFLION(J),ONN          00234900
      WRITE(T,9320)HQTYTO(3),LFLION(J),LFLROW(3),LFLION(J),ONN          00235000
      WRITE(T,9330)HQTYTO(3),LFLION(J),LMTROW(1),ONE                    00235100
      WRITE(T,9330)HQTYTO(3),LFLION(J),CSTROW(1),LFLVCP(J)              00235200
      WRITE(T,9330)HQTYTO(3),LFLION(J),CSTROW(2),LFLVOP(J)              00235300
      LFLVAL=LFLVCP(J)+LFLVOP(J)                                        00235400
      WRITE(T,9330)HQTYTO(3),LFLION(J),CSTROW(5),LFLVAL                 00235500
 330  CONTINUE                                                          00235600
C                                                                       00235700
C                                                                       00235800
C                                                                       00235900
C              BYPASS IF NO RESOURCE RECOVERY                           00236000
C                                                                       00236100
 335  IF(NRESRC.EQ.0) GO TO 400                                         00236200
      DO 340 J=1,NRESRC                                                 00236300
      NMKTRJ=NUMMKT(J)                                                  00236400
      DO 340 K=1,NMKTRJ                                                 00236500
      WRITE(T,9350)WQTYTO(4),MKTION(J,K),MKTROW(1),MKTION(J,K),ONE
      WRITE(T,9350)WQTYTO(4),MKTION(J,K),MKTROW(2),MKTION(J,K),ONE
      WRITE(T,9360)WQTYTO(4),MKTION(J,K),CSTROW(4),PRICET(J,K)
      WRITE(T,9360)WQTYTO(4),MKTION(J,K),CSTROW(5),PRICET(J,K)
      WRITE(T,9360)WQTYTO(4),MKTION(J,K),CHGROW(4),PRICET(J,K)
 340  CONTINUE
 9350 FORMAT(4X,A4,I3,3X,A5,I3,2X,F12.6)
 9360 FORMAT(4X,A4,I3,3X,A8,2X,F12.6)
C
C     ***    WRITE MARKER CARD AT BEGINNING OF INTEGER VARIABLES
C     ***    CHANGED BY LJ (&)
C
 400  CONTINUE
 9400 FORMAT(4X,6HDELTAS,4X,8H"MARKER",17X,8H"INTORG")
C
C
C              BYPASS IF NO TRANSFER STATIONS
C
      IF(NUMTRF.EQ.0) GO TO 415
      DO 410 J=1,NUMTRF
      WRITE(T,9321)DELTAS(1),TRFION(J),TRFROW(2),TRFION(J),TRFLLT(J)
      WRITE(T,9321)DELTAS(1),TRFION(J),TRFROW(3),TRFION(J),TRFULT(J)
      WRITE(T,9331)DELTAS(1),TRFION(J),LMTROW(2),ONE
      WRITE(T,9331)DELTAS(1),TRFION(J),CSTROW(1),TRFFCP(J)
      WRITE(T,9331)DELTAS(1),TRFION(J),CSTROW(2),TRFFOP(J)
      TRFFAL=TRFFCP(J)+TRFFOP(J)
      WRITE(T,9331)DELTAS(1),TRFION(J),CSTROW(5),TRFFAL
 410  CONTINUE
 9321  FORMAT(4X,A4,I3,3X,A5,I3,2X,F12.1)
 9331 FORMAT (4X,A4,I3,3X,A8,2X,F12.1)
C
C
C              BYPASS IF NO PLANTS
C
 415  IF(NUMPLT.EQ.0) GO TO 425
      DO 420 J=1,NUMPLT
      WRITE(T,9321)DELTAS(2),PLTION(J),PLTROW(2),PLTION(J),PLTLLT(J)
      WRITE(T,9321)DELTAS(2),PLTION(J),PLTROW(3),PLTION(J),PLTULT(J)
      WRITE(T,9331)DELTAS(2),PLTION(J),LMTROW(3),ONE
      WRITE(T,9331)DELTAS(2),PLTION(J),CSTROW(1),PLTFCP(J)
      WRITE(T,9331)DELTAS(2),PLTION(J),CSTROW(2),PLTFOP(J)
      PLTFAL=PLTFCP(J)+PLTFOP(J)
      WRITE(T,9331)DELTAS(2),PLTION(J),CSTROW(5),PLTFAL
      WRITE(T,9331)DELTAS(2),PLTION(J),CHGROW(1),PLTFCP(J)
      WRITE(T,9331)DELTAS(2),PLTION(J),CHGROW(2),PLTFOP(J)
 420  CONTINUE
C
C     ***    MAYBE WRITE MARKER CARD AT END OF INTEGER VARIABLES
C     ***    THIS PART DELETED BY LJ(&)
 425  CONTINUE
 9401 FORMAT(4X,6HDELTAE,4X,8H"MARKER",17X,8H"INTEND")
C
      DO 430 J=1,NUMLFL
      WRITE(T,9321)DELTAS(3),LFLION(J),LFLROW(2),LFLION(J),LFLLLT(J)
      WRITE(T,9321)DELTAS(3),LFLION(J),LFLROW(3),LFLION(J),LFLULT(J)
      WRITE(T,9331)DELTAS(3),LFLION(J),LMTROW(4),ONE
      WRITE(T,9331)DELTAS(3),LFLION(J),CSTROW(1),LFLFCP(J)
      WRITE(T,9331)DELTAS(3),LFLIDN(J),CSTROW(2),LFLFOP(J)
      LFLFAL=LFLFCP(J)+LFLFOP(J)
      WRITE(T,9331)DELTAS(3),LFLIDN(J),CSTROW(5),LFLFAL
 430  CONTINUE
C
C     ***     MAYBE WRITE MARKER CARD AT END OF INTEGER VARIABLES
C     ***     THIS PART CHANGED BY LJ (6), NOT NEEDED FOR CDC
C
C
C
C     *** WRITE RHS SECTION ***********************************
C
      WRITE(T,9510)
 9510 FORMAT("RHS")
C
 9520 FORMAT(4X,A8,2X,A5,I3,2X,F12.1)
 9530 FORMAT(4X,A8,2X,A5,I3,2X,F12.1)
 9540 FORMAT(4X,A8,2X,A8,2X,F12.1)
      DO 510 J=1,NUMSRC
      WRITE(T,9520)RHSNAM(1),SRCROW,SRCIDN(J),SRCTOT(J)
 510  CONTINUE
C
C
C             BYPASS IF NO RESOURCE RECOVERY
C
      IF(NRESRC.EQ.0) GO TO 525
      DO 520 J=1,NRESRC
      NMKTRJ=NUMMKT(J)
      DO 520 K=1,NMKTRJ
      WRITE(T,9530)RHSNAM(1),MKTROW(2),MKTIDN(J,K),DEMAND(J,K)
 520  CONTINUE
 525  WRITE(T,9540)RHSNAM(1),LMTROW(1),LTOLFL
      WRITE(T,9540)RHSNAM(1),LMTROW(2),LNUMTR
      WRITE(T,9540)RHSNAM(1),LMTROW(3),LNUMPL
      WRITE(T,9540)RHSNAM(1),LMTROW(4),LNUMLF
      WRITE(T,9540)RHSNAM(1),CSTROW(1),LCAPTL
C
C
C         WRITE RHS FOR CHANGE ROWS
C
C
      DO 530 J=1,NUMSRC
 530  WRITE(T,9520)RHSNAM(2),SRCROW,SRCIDN(J),SRCTOT(J)
C
C
C               BYPASS IF NO RESOURCE RECOVERY
C
      IF(NRESRC.EQ.0) GO TO 550
      DO 540 J=1,NRESRC
      NMKTRJ=NUMMKT(J)
      DO 540 K=1,NMKTRJ
      WRITE(T,9530)RHSNAM(3),MKTROW(2),MKTIDN(J,K),DEMAND(J,K)
 540  CONTINUE
 550  WRITE(T,9540)RHSNAM(4),LMTROW(1),LTOLFL
C
      WRITE(T,9540)RHSNAM(5),CSTROW(1),LCAPTL
C
```

```
C
C     ***     WRITE BOUNDS SECTION    **************************
      WRITE(T,9610)
 9610 FORMAT("BOUNDS")
C
 9620 FORMAT(1X,A2,1X,A8,2X,A4,I3,3X,F12.1)
C
C
C
C              BYPASS IF NO TRANSFER STATIONS
C
      IF(NUMTRF.EQ.0) GO TO 615
      DO 610 J=1,NUMTRF
      STATJ=TRSTAT(J)+1
  610 WRITE(T,9620)FXUP(STATJ),BNDNAM,DELTAS(1),TRFION(J),FXUPV(STATJ)
C
C
C              BYPASS IF NO PLANTS
C
  615 IF(NUMPLT.EQ.0) GO TO 625
      DO 620 J=1,NUMPLT
      STATJ=PLSTAT(J)+1
  620 WRITE(T,9620)FXUP(STATJ),BNDNAM,DELTAS(2),PLTION(J),FXUPV(STATJ)
  625 DO 630 J=1,NUMLFL
      STATJ=LFSTAT(J)+1
  630 WRITE(T,9620)FXUP(STATJ),BNDNAM,DELTAS(3),LFLION(J),FXUPV(STATJ)
C
      WRITE(T,9700)
 9700 FORMAT("ENDATA")
C
      ENDFILE T
      WRITE(W,9800)
 9800 FORMAT("-","          WRITE TO T FILE COMPLETED.")
      RETURN
      END
      SUBROUTINE OSETUP
C
C       ****************************************************************
C       *
C       *    THIS ROUTINE WRITES TO DEVICE O ALL PARAMETERS AND NAMES *
C       *    READ IN BY THE EDIT AND SETUP PROGRAM AND NEEDED FOR      *
C       *    THE OUTPUT REPORT                                         *
C       *                                                              *
C       ****************************************************************
C
C     ***    INITIAL DECLARATION OF L VARIABLES AS REAL
C
      IMPLICIT REAL(L)
C
C     ***    DECLARATION OF VARIABLES FOR NAMES AND
C            TITLES
      REAL ASTRSK
      REAL TITLE,RUNNAM,VECNAM,SRCNAM,TRFNAM,PLTNAM,
     1         LFLNAM,RSCNAM,MKTNAM
C
      INTEGER R,W,T,O
      INTEGER TOTERR,ERFLAG,RUNYRS,
     1              RUNBEG,RUNEND,INREP,OUTREP,TWRITE,OWRITE,LANDCD,
     2               SRCIDN,TRFIDN,PLTIDN,LFLIDN,MKTIDN,
     3              ORIGID,DESTID,TRSTAT,PLSTAT,LFSTAT,STATJ
C
C     *** GENERAL COMMON BLOCK
C
      COMMON/GENDAT/ASTRSK(15),TITLE(5),RUNNAM(5),
     1                   LTOLFL,LNUMTR,LNUMPL,LNUMLF,LCAPTL,RATE,
     2                   R,W,T,O,TOTERR,ERFLAG,DAT,
     3                   RUNBEG,RUNEND,RUNYRS,INREP,OUTREP,TWRITE,OWRITE,
     4                   LANDCD,NUMVEC,NUMSRC,NUMTRF,NUMPLT,NUMLFL,NRESRC,
     5                   NUMMKT(9),NPAGE,LINPAG
C
C     ***    VEHICLES COMMON BLOCK
C
      COMMON/VECDAT/VECNAM(8),COSTTH(8)
C
C     ***    SOURCES COMMON BLOCK
C
      COMMON/SRCDAT/SRCNAM(200),SRCTOT(200),ALWAST,SRCIDN(200)
C
C     ***    TRANSFER STATIONS COMMON BLOCK
C
      COMMON/TRFDAT/TRFNAM(20),TRFLLT(20),TRFULT(20),
     1                   TRFFCP(20),TRFVCP(20),TRFFOP(20),TRFVOP(20),
     2                   TRFIDN(20),TRSTAT(20)
C
C     ***    COMMON FOR PLANTS
C
      COMMON/PLTDAT/PLTNAM(20),PLTLLT(20),PLTULT(20),
     1                   PLTFCP(20),PLTVCP(20),PLTFOP(20),PLTVOP(20),
     2                   PLTREP(20),PLTRRN(20,9),
     3                   PLTIDN(20),PLSTAT(20)
C
C     ***    COMMON FOR LANDFILLS
```

```
C
      COMMON/LFLDAT/LFLNAM(20),LFLLLT(20),LFLULT(20),
     1              LFLFCP(20),LFLVCP(20),LFLFOP(20),LFLVOP(20),
     2              LFLIDN(20),LFSTAT(20)
C
C     ***   COMMON FOR MARKETS
      COMMON/MKTDAT/RSCNAM(9),MKTNAM(9,9),DEMAND(9,9),PRICET(9,9),
     1                                              MKTIDN(9,9)
C
C     ***   COMMON FOR TRANSPORTATION ARCS
C
      COMMON/LNKDAT/ARCOST(5000),ORIGID(5000),DESTID(5000),
     1              NANDST,NANDSP,NANDSL,NANDTP,NANDTL,NANDPL,NANDPM
C
      IF(INREP.GT.0) CALL HEADER(9)
C
C  WRITE AND FORMAT CODES WITH COMMENT MARK ARE FOR FORMATTED
C  FILE D - TRANSFER. THIS WILL BE USED WHEN THE SETUP PROGRAM IS
C  RUN ON IBM AND THE DATA TRANSFERED VIA 9 TRACK TAPE. IN THIS
C  INSTANCE REMOVE C"S AND THE NONFORMATED READS.
C
C    WRITE (D,1000) TITLE,RUNNAM,
     WRITE (D) TITLE,RUNNAM,
     1         RUNBEG,RUNEND,OUTREP,
     2         NUMVEC,NUMSRC,NUMTRF,NUMPLT,NUMLFL,NRESPC,NUMMKT,
     3         RATE,
     4          VECIDN,VECNAM,COSTTH,
     5          SRCIDN,SRCNAM,ALWAST
C     WRITE (D,1010) TRFIDN,TRFNAM,TRFFCP,TRFVCP,TRFFOP,TRFVOP
C     WRITE (D,1010) PLTIDN,PLTNAM,PLTFCP,PLTVCP,PLTFOP,PLTVOP
C     WRITE (D,1010) LFLIDN,LFLNAM,LFLFCP,LFLVCP,LFLFOP,LFLVOP,
C                    RSCNAM,MKTIDN,MKTNAM,PRICET
C     WRITE (D,1030) NANDST,NANDSP,NANDSL,NANDTP,NANDTL,NANDPL,NANDPM,
C    1         ORIGID,DESTID,ARCOST
      WRITE (D) TRFIDN,TRFNAM,TRFFCP,TRFVCP,TRFFOP,TRFVOP
      WRITE (D) PLTIDN,PLTNAM,PLTFCP,PLTVCP,PLTFOP,PLTVOP
      WRITE (D) LFLIDN,LFLNAM,LFLFCP,LFLVCP,LFLFOP,LFLVOP,
     9          RSCNAM,MKTIDN,MKTNAM,PRICET
      WRITE (D) NANDST,NANDSP,NANDSL,NANDTP,NANDTL,NANDPL,NANDPM,
     1         ORIGID,DESTID,ARCOST
      ENDFILE D
     WRITE(W,500)
 500 FORMAT("-","  WRITE OF FILE D COMPLETED.")
C1000 FORMAT (5A8/5A8/9I5/9I5/F10.5/9A8/8F10.4/13(15I5/),5I5/
C    1      20(10A8/),F12.0)
C1010 FORMAT (10I5/10I5/10A8/10A8/6F12.1/6F12.1/6F12.1/2F12.1/
C    1  8F10.5/8F10.5/4F10.5/8F10.1/8F10.1/4F10.1/
C    2  8F10.5/8F10.5/4F10.5)
C1020 FORMAT (9A8/9(9I5/),9(9A8/),9(5F15.4/4F15.4/))
C1030 FORMAT (7I5/62(16I5/),8I5/62(16I5/),8I5/142(7F11.5/),
C    1  6F11.5)
     RETURN
     END
```

APPENDIX 8

SOURCE CODE OF RWP FOR CDC

```
      PROGRAM REPORT(TAPE12,TAPE5,TAPE6,OUTPUT,TAPE6=OUTPUT)
C MODIF. AUG.20, 1980 - JP
C
C  ******************************************************************
C  *                                                                *
C  *              REPORT GENERATION - MAIN PROGRAM                  *
C  *                                                                *
C  *                                                                *
C  *                 TAPE12 = UNIT WITH T DATA - DATA FROM APEX     *
C  *                 TAPE5 = UNIT FOR MAIN PROG. INPUT CARD         *
C  *                 TAPE8 = UNIT WITH D DATA - DATA FROM SETUP PR.*
C  *                 TAPE6 = UNIT FOR PRINTOUT                      *
C  *                                                                *
C  ******************************************************************
C
C
      INTEGER FACTYP,LINPAG
      REAL     GENNAM,FACNAM,TITLE,RUNNAM
      COMMON/GEN/NOSUBT,IDSUBT(40)
      COMMON /NUMBS/ NPAGE,NLINES,LINPAG,
     1               RATE,NUMYRS,INREP,OUTREP,
     2               NUMVEC,NUMSRC,NUMTRF,NUMPLT,NUMLFL,NRM,NRESRC,
     3               NUMMKT(9)
      COMMON/HEAD/TITLE(5),RUNNAM(5),RUNBEG,RUNEND,DAT
      LINPAG=60
      NPAGE=1
      CALL DATE(DAT)
      CALL DFETCH
      CALL TFETCH
      CALL FXSUM
      IF (OUTREP.EQ.0) STOP
      CALL SUMREP
      READ(5,100) GENNAM,FACNAM,FACTYP
 100  FORMAT(2A8,A4)
      READ(5,101) NOSUBT,(IDSUBT(N),N=1,NOSUBT)
 101  FORMAT(16I5)
      CALL DETGEN(GENNAM)
      CALL DETFAC(FACNAM,FACTYP)
      IF (NRESRC.GT.0) CALL MKTREP
      CALL TPSHIP
      STOP
      END
```

```
C
C
      SUBROUTINE TFETCH
C
C
C   ****************************************************************
C   *                                                              *
C   *  THIS SUBROUTINE FETCHES THE DATA PREPARED IN THE EDIT-AND-  *
C   *  SETUP PROGRAM (THE FILE PREPARED BY DSETUP) AND BY THE MILP *
C   *  PACKAGE (FORTFILE).                                         *
C   *                                                              *
C   ****************************************************************
C
C
      INTEGER L,M,N,P,LINPAG
      INTEGER RNAM(6)
      INTEGER TYPE(60),R,  T,D,ORIGID,DESTID,OUTREP
      DIMENSION W(7),IPM(7),IW(7),IPARTS(9)
      EQUIVALENCE (IW(1),W(1))
      COMMON/NUMBS/NPAGE,NLINES,LINPAG,
     1             RATE,NUMYRS,INREP,OUTREP,
     2             NUMVEC,NUMSRC,NUMTRF,NUMPLT,NUMLFL,NRM,NRESRC,
     3             NUMMKT(9)
       COMMON/ARCDAT/ARCOST(5000),ORIGID(5000),DESTID(5000),
     1        NANDST,NANDSP,NANDSL,NANDTP,NANDTL,NANDPL,NANDPM,NANDWL
      COMMON/OUT/ARCWST(5000),WST(20,3),WM(81),DELTA(20,3),COST(6),DISC
      DATA RNAM(1)/"CAPITALC"/,
     1  RNAM(2)/"OPERATNG"/,RNAM(3)/"TRANSPRT"/,RNAM(5)/"REVENUES"/,
     2  RNAM(6)/"NETCOST "/,RNAM(4)/"TOTCOSTS"/
      NANDWT=NANDPM+NUMTRF
      NANDWP=NANDWT+NUMPLT
      NANDWL=NANDWP+NUMLFL
      NRM=0
      IF(NRESRC.EQ.0) GO TO 107
      DO100 NR=1,NRESRC
100   NRM=NRM+NUMMKT(NR)
107   NANDWM=NANDWL+NRM
      NANDDT=NANDWM+NUMTRF
      NANDDP=NANDDT+NUMPLT
      NANDDL=NANDDP+NUMLFL
      REWIND 12
      READ(12)(W(K),K=1,7)
      DO 10 J=5,7
      IPM(J)="P"
      IF(W(J).LT.0.)IPM(J)="M"
      XJIM=W(J)
10    W(J)=ABS(XJIM)
      IF (OUTREP.EQ.2)
     1WRITE(6,101)(W(K),K=1,4),(IPM(K),W(K),K=5,7)
101   FORMAT(4A10,2(A1,F9.3),A1,F19.9)
      READ(12)(W(K),K=1,7)
      DO 20 J=4,5
      IPM(J)="P"
      IF(W(J).LT.0.)IPM(J)="M"
      XJIM=W(J)
20    W(J)=ABS(XJIM)
      NROWS=IW(6)
      NCOLS=IW(7)
      IF (OUTREP.EQ.2)
     1WRITE(6,102)(W(K),K=1,3),(IPM(K),W(K),K=4,5),(IW(K),K=6,7)
  102 FORMAT(3A10,2(A1,F9.3),2I10)
      IF(NROWS.LE.0)GO TO 300
C
C         GET THE ROW AND COLUMN SECTIONS
C     ****************************************
C
      MINROW=NROWS-8
      MAXROW=MINROW+4
      COST(4)=0.
      DO 200 I=1,NROWS
      READ(12)(W(K),K=1,2)
      IF(IW(1).EQ."$$END$$")GO TO 999
      IF(I.LT.MINROW.OR.I.GT.MAXROW) GO TO 200
      DO240 J=1,5
      IF(J.LE.3) JP=J
      IF(J.GT.3) JP=J+1
      IF(IW(1).EQ.RNAM(JP)) GO TO 241
  240 CONTINUE
  241 COST(JP)=W(2)
      IF(JP.LE.3) COST(4)=COST(4)+COST(JP)
      IF(JP.EQ.5.AND.OUTREP.EQ.2) WRITE(6,26) RNAM(4),COST(4),COST(4)
      IF (OUTREP.EQ.2) WRITE(6,26) W(1),W(2),COST(JP)
   26 FORMAT(1X,A12,2D20.8)
  200 CONTINUE
  300 IF(NCOLS.LE.0)GO TO 998
      DO 400 I=1,NCOLS
      READ(12)(W(K),K=1,2)
      IF(IW(1).EQ."$$END$$")GO TO 999
      IF(I.GT.NANDPM) GO TO 35
      ARCWST(I)=W(2)
      GO TO 242
   35 IF(I.GT.NANDWT) GO TO 36
      NDIF=I-NANDPM
      WST(NDIF,1)=W(2)
      GO TO 242
   36 IF(I.GT.NANDWP) GO TO 37
      NDIF=I-NANDWT
      WST(NDIF,2)=W(2)
      GO TO 242
   37 IF(I.GT.NANDWL) GO TO 380
      NDIF=I-NANDWP
      WST(NDIF,3)=W(2)
      GO TO 242
  380 IF(I.GT.NANDWM) GO TO 38
      NDIF=I-NANDWL
      WM(NDIF)=W(2)
      GO TO 242
   38 IF(I.GT.NANDDT) GO TO 39
      NDIF=I-NANDWM
      DELTA(NDIF,1)=W(2)
      GO TO 242
   39 IF(I.GT.NANDDP) GO TO 40
      NDIF=I-NANDDT
      DELTA(NDIF,2)=W(2)
      GO TO 242
 40   IF(I.GT.NANDOOL) GO TO 41
      NDIF=I-NANDDP
      DELTA(NDIF,3)=W(2)
 242  IF (OUTREP.EQ.2) WRITE(6,26) W(1),W(2)
 400  CONTINUE
C
C
C      ALL T DATA NOW IN VECTORS. 41 MARKS THE POINT AT WHICH
C      CALCULATIONS MAY BEGIN, AS LONG AS I=NCOLS AT THIS TIME.
C
 41   GOTO 42
C      IF (OUTREP.EQ.2) WRITE(6,941) I,NCOLS
C941  FORMAT(" ","I=",I6," NCOLS=",I6,".  SHOULD BE EQUAL.(41)")
C     CALL EXIT
 42   CONTINUE
 998  READ(12)(W(K),K=1,7)
      IF(IW(1).EQ."$$END$$")GO TO 999
      IF (OUTREP.EQ.2) WRITE(6,997)
 997  FORMAT(" END OF FILE NOT REACHED")
      CALL EXIT
 999  RETURN
      END
```

```
C
C
C
      SUBROUTINE DFETCH
C
C        ***************************************************************
C        *                                                             *
C        *   THIS ROUTINE READS FROM DEVICE 8 ALL PARAMETERS AND NAMES *
C        *   PREPARED BY THE EDIT AND SETUP PROGRAM AND NEEDED FOR     *
C        *   THE OUTPUT REPORT                                         *
C        *                                                             *
C        ***************************************************************
C
C      ***   INITIAL DECLARATION OF L VARIABLES AS REAL,
C      ***   ALL IMPLICIT INTEGER VARIABLES AS INTEGER
C
       IMPLICIT REAL(A-H,L,O-Z),INTEGER(I-K,M,N)
C
C      ***   DECLARATION OF REAL VARIABLES (FOR TITLES AND
C      ***                     NAMES)
       REAL TITLE,RUNNAM,VECNAM,SRCNAM,TRFNAM,PLTNAM,
     1       LFLNAM,RSCNAM,MKTNAM
C
       INTEGER T,D,K,W,LINPAG
      INTEGER  RUNBEG,RUNEND,INREP,OUTREP,
     1           VECIDN,SRCIDN,TRFIDN,PLTIDN,LFLIDN,MKTIDN,
     2           ORIGID,DESTID
C
C      *** GENERAL COMMON BLOCK
C
      COMMON/HEAD/TITLE(5),RUNNAM(5),RUNBEG,RUNEND,DAT
      COMMON/NUMBS/NPAGE,NLINES,LINPAG,
     1               RATE,NUMYRS,INREP,OUTREP,
     2               NUMVEC,NUMSRC,NUMTRF,NUMPLT,NUMLFL,NRM,NRESRC,
     3               NUMMKT(9)
C
C      ***   VEHICLES COMMON BLOCK
C
       COMMON/VECDAT/VECNAM(8),COSTTH(8),VECIDN
C
C      ***   SOURCES COMMON BLOCK
C
       COMMON/SRCDAT/SRCNAM(200),SRCTOT(200),SRCIDN(200),ALWAST
C
C      ***   FACILITIES COMMON BLOCK
C
      COMMON/FACDAT/TRFNAM(20),PLTNAM(20),LFLNAM(20),TRFIDN(20),
     1  PLTIDN(20),LFLIDN(20),TRFFCP(20),PLTFCP(20),LFLFCP(20),
     2  TRFVCP(20),PLTVCP(20),LFLVCP(20),TRFFOP(20),PLTFOP(20),
     3  LFLFOP(20),TRFVOP(20),PLTVOP(20),LFLVOP(20)
C
C      ***   COMMON FOR MARKETS
       COMMON/MKTDAT/RSCNAM(9),MKTNAM(9,9),MKTIDN(9,9),DEMAND(9,9),
     1               PRICET(9,9)
C
C      ***   COMMON FOR TRANSPORTATION ARCS
C
```

```
       COMMON/ARCDAT/ARCOST(5000),ORIGID(5000),DESTID(5000),
     1           NANDST,NANDSP,NANDSL,NANDTP,NANDTL,NANDPL,NANDPM,NANDWL
C
      REWIND 8
      READ(8)   TITLE,RUNNAM,
     1          RUNBEG,RUNEND,OUTREP,
     2          NUMVEC,NUMSRC,NUMTRF,NUMPLT,NUMLFL,NRESRC,NUMMKT,
     3          RATE,
     4          VECION,VECNAM,COSTTH,
     5          SRCION,SRCNAM,ALWAST
      READ(8) TRFIDN,TRFNAM,TRFFCP,TRFVCP,TRFFOP,TRFVOP
      READ(8) PLTIDN,PLTNAM,PLTFCP,PLTVCP,PLTFOP,PLTVOP
      READ(8)   LFLIDN,LFLNAM,LFLFCP,LFLVCP,LFLFOP,LFLVOP,
     9          RSCNAM,MKTIDN,MKTNAM,PRICET
      READ(8)   NANDST,NANDSP,NANDSL,NANDTP,NANDTL,NANDPL,NANDPM,
     1          ORIGID,DESTID,ARCOST
                NUMYRS=RUNEND-RUNBEG+1
      IF (OUTREP.NE.2) RETURN
      WRITE(6,100)   TITLE,RUNNAM,
     1          RUNBEG,RUNEND,OUTREP,
     2          NUMVEC,NUMSRC,NUMTRF,NUMPLT,NUMLFL,NRESRC,NUMMKT,
     3          RATE
      WRITE(6,101) VECION
      IF(NUMVEC.NE.0)WRITE(6,102) (VECNAM(N),N=1,NUMVEC)
      WRITE(6,103) (COSTTH(I),I=1,NUMVEC)
      IF(NUMSRC.NE.0) WRITE(6,101) ( SRCIDN(N),N=1,NUMSRC)
      IF(NUMSRC.NE.0) WRITE(6,102) (SRCNAM(N),N=1,NUMSRC)
      WRITE(6,103) ALWAST
      IF(NUMTRF.NE.0) WRITE(6,101) ( TRFIDN(N),N=1,NUMTRF)
      IF(NUMTRF.NE.0) WRITE(6,102) (TRFNAM(N),N=1,NUMTRF)
      IF(NUMTRF.NE.0) WRITE(6,103) (TRFFCP(N),N=1,NUMTRF),(TRFVCP(N),
     1     N=1,NUMTRF),(TRFFOP(N),N=1,NUMTRF),(TRFVOP(N),N=1,NUMTRF)
      IF(NUMPLT.NE.0) WRITE(6,101) ( PLTIDN(N),N=1,NUMPLT)
      IF(NUMPLT.NE.0)WRITE(6,102) (PLTNAM(N),N=1,NUMPLT)
      IF(NUMPLT.NE.0) WRITE(6,103) (PLTFCP(N),N=1,NUMPLT),(PLTVCP(N),
     1     N=1,NUMPLT),(PLTFOP(N),N=1,NUMPLT),(PLTVOP(N),N=1,NUMPLT)
      IF(NUMLFL.NE.0) WRITE(6,101) ( LFLIDN(N),N=1,NUMLFL)
      IF(NUMLFL.NE.0) WRITE(6,102) (LFLNAM(N),N=1,NUMLFL)
      IF(NUMLFL.NE.0) WRITE(6,103) (LFLFCP(N),N=1,NUMLFL),(LFLVCP(N),
     1     N=1,NUMLFL),(LFLFOP(N),N=1,NUMLFL),(LFLVOP(N),N=1,NUMLFL)
      IF(NRESRC.EQ.0) GO TO 2
      DO1 N=1,NRESRC
      NMKT=NUMMKT(N)
      IF(NMKT.NE.0) WRITE(6,101) (MKTIDN(N,NM),NM=1,NMKT)
      IF(NMKT.NE.0) WRITE(6,102) (MKTNAM(N,NM),NM=1,NMKT)
      IF(NMKT.NE.0) WRITE(6,103) (PRICET(N,NM),NM=1,NMKT)
 1    CONTINUE
 2    CONTINUE
      WRITE(6,101)   NANDST,NANDSP,NANDSL,NANDTP,NANDTL,NANDPL,NANDPM
      WRITE(6,101) (ORIGID(N),N=1,NANDPM)
      WRITE(6,101) (DESTID(N),N=1,NANDPM)
      WRITE(6,103) (ARCOST(N),N=1,NANDPM)
100   FORMAT(T4,2(5A8)/4X,3I4/8X,15I4/4X,F6.2)
101   FORMAT(T4,20I4)
102   FORMAT(T4,10A8)
103   FORMAT(T4,10F13.2)
                RETURN

                END
```

```
C
C
C
      SUBROUTINE HEADER(NOHEAD,NCONT)
C
C
C
      REAL TITLE,RUNNAM
      INTEGER RUNBEG,RUNEND,DAT,LINPAG
      COMMON/HEAD/TITLE(5),RUNNAM(5),RUNBEG,RUNEND,DAT
      COMMON/NUMBS/NPAGE,NLINES,LINPAG,
     1                    RATE,NUMYRS,INREP,OUTREP,
     2                    NUMVEC,NUMSRC,NUMTRF,NUMPLT,NUMLFL,NRM,NRESRC,
     3                    NUMMKT(9)
      WRITE(6,500) TITLE,DAT,NPAGE
      NPAGE=NPAGE+1
      NLINES=1
      IF(NOHEAD.EQ.1) WRITE(6,1)
      IF(NOHEAD.EQ.2) WRITE(6,2)
      IF(NOHEAD.EQ.3) WRITE(6,3)
      IF(NOHEAD.EQ.4) WRITE(6,4)
      IF(NOHEAD.EQ.5) WRITE(6,5)
      IF (NOHEAD.EQ.6) WRITE (6,6)
      IF (NCONT.EQ.1) WRITE (6,7)
      WRITE(6,502) RUNNAM,RUNBEG,RUNEND
      NLINES=7
  500 FORMAT("1","ONTARIO MINISTRY OF THE ENVIRONMENT, WASTE MANAGE",
     1 "MENT SYSTEMS MODEL",3X,5A8,3X,A10,2X,"PAGE",I3)
    1 FORMAT("0","EXECUTIVE SUMMARY")
    2 FORMAT("0","SUMMARY REPORT OF WASTE GENERATION AREA SHIPMENTS")
    3 FORMAT("0","DETAILED REPORT OF WASTE GENERATION AREA SHIPMENTS")
    4 FORMAT("0","DETAILED REPORT OF FACILITY OPERATIONS")
    5 FORMAT("0","MARKETING REPORT")
    6 FORMAT ("0","DETAILED REPORT ON TRANSFER STATIONS AND PLANTS",
     1 " SHIPMENTS")
    7 FORMAT (     5X," - CONTINUATION - ")
  502 FORMAT("0","IDENTIFICATION OF THE RUN: ",5A8//" TIME PERIOD OF ",
     1  "THE RUN:   ",I4," TO ",I4)
      RETURN
      END
```

```
C
C
C
      SUBROUTINE EXSUM
C
C
C  ***********************************************************
C  *                                                         *
C  *  THIS SUBROUTINE WRITES THE EXECUTIVE SUMMARY REPORT.   *
C  *                                                         *
C  ***********************************************************
C
C
C
C
C        ***   INITIAL DECLARATION OF L VARIABLES AS REAL,
C        ***   ALL IMPLICIT INTEGER VARIABLES AS INTEGER
C
        IMPLICIT REAL(A-H,L,O-Z),INTEGER(I-K,M,N)
C
C        ***   DECLARATION OF REAL VARIABLES (FOR TITLES AND
C        ***                        NAMES)
        REAL TITLE,RUNNAM,VECNAM,SRCNAM,TRFNAM,PLTNAM,
     1        LFLNAM,RSCNAM,MKTNAM,COSTN(2,6),WASTN(2,3),TOTOF(2),
     2        FACIL(2),PERCNT(2),SYSTOT(2)
       REAL TNAM,PNAM,LNAM,MNAM,BLANK8
       REAL   AVCOST(6),UNCOST(6),PERCST(6),VOL(3),AVVOL(3),
     1       PERVOL(3),TUSE,PUSE,LUSE,MUSE,DT(20),DP(20),DL(20),
     1        CAP(3),OP(3),TR(3),TOT(3),PER(3)
       REAL BLANK4,YES,NO
C
        INTEGER T,D,R,W,LINPAG
       INTEGER  RUNBEG,RUNEND,INREP,OUTREP,
     1          VECIDN,SRCIDN,TRFIDN,PLTIDN,LFLIDN,MKTIDN,
     2          ORIGID,DESTID
C
C        *** GENERAL COMMON BLOCK
C
       COMMON/NUMRS/NPAGE,NLINES,LINPAG,
     1              RATE,NUMYRS,INREP,OUTREP,
     2              NUMVEC,NUMSRC,NUM(4),NRESRC,
     3              NUMMKT(9)
C
C        ***   VEHICLES COMMON BLOCK
C
        COMMON/VECDAT/VECNAM(8),COSTTH(8),VECIDN
C
C        ***   SOURCES COMMON BLOCK
C
        COMMON/SRCDAT/SRCNAM(200),SRCTOT(200),SRCIDN(200),ALWAST
C
C        ***   FACILITIES COMMON BLOCK
C
       COMMON/FACDAT/TRFNAM(20),PLTNAM(20),LFLNAM(20),IDN(20,3),
     1              FCP(20,3),VCP(20,3),FOP(20,3),VOP(20,3)
C
C        ***   COMMON FOR MARKETS
```

```
       COMMON/MKTDAT/RSCNAM(9),MKTNAM(9,9),MKTIDN(9,9),DEMAND(9,9),
     1              PRICET(9,9)
C
C      ***   COMMON FOR TRANSPORTATION ARCS
C
       COMMON/ARCDAT/ARCOST(5000),ORIGID(5000),DESTID(5000),
     1        NANDST,NANDSP,NANDSL,NANDTP,NANDTL,NANDPL,NANDPM,NANDWL
C
      COMMON/OUT/ARCWST(5000),WST(20,3),WM(81),DELTA(20,3),COST(6),DISC
      EQUIVALENCE (DELTA(1,1),DT(1)),(DELTA(1,2),DP(1)),
     1            (DELTA(1,3),DL(1)),(NUM(4),NRM)
C
      DATA COSTN/"CAPITAL ","COST    ","OPERATIN","G COST  ",
     1           "TRANSPOR","T COST  ","TOTAL CO","ST      ",
     2           "REVENUE ","        ","NET PROF","IT      "/,
     3     WASTN/"WASTE GE","NERATED ","WASTE RE","COVERED ",
     4           "WASTE DI","SPOSED  "/,
     5     TOTOF/"TOTAL CO","ST OF FA"/,
     6     FACIL/"CILITIES","        "/,PERCNT/"PERCENT ","OF SYSTE"/,
     7     SYSTOT/"M TOTAL ","        "/,BLANK4/"    "/,YES/" YES"/,
     8     NO/" NO "/,BLANK8/"        "/
C
C
C         CALCUATE DISCOUNT FACTOR AND SAVE IN COMMON/OUT/
C
C
      IF (RATE.EQ.0) DISC=RUNYRS
      IF (RATE.EQ.0) GO TO 20
      DISC=NUMYRS/((1.+RATE)**NUMYRS-1.)*RATE*(1.+RATE)**NUMYRS
C
C
C
  20  CALL HEADER(1,0)
      WRITE(6,504)
      NLINES=NLINES+8
      VOL(1)=ALWAST
      VOL(2)=0.
      IF(NRM.EQ.0) GO TO 100
      DO10 I=1,NRM
 10   VOL(2)=VOL(2)+WM(I)
 100  VOL(3)=VOL(1)-VOL(2)
      COST(4)=0.
      COST(5)=-COST(5)*DISC
      DO11 I=1,6
      IF(I.EQ.2.OR.I.EQ.3) COST(I)=COST(I)*DISC
      IF(I.LT.4) COST(4)=COST(4)+COST(I)
      IF(I.EQ.6) COST(6)=COST(5)-COST(4)
      AVCOST(I)=COST(I)/NUMYRS
      UNCOST(I)=COST(I)/ALWAST
      IF(I.GT.3) GO TO 11
      AVVOL(I)=VOL(I)/NUMYRS
      PERVOL(I)=VOL(I)/ALWAST*100.
 11   CONTINUE
      DO 12 I=1,4
 12   PERCST(I) = COST(I)/COST(4)*100.
      WRITE(6,505) ((COSTN(J,I),J=1,2),COST(I),AVCOST(I),UNCOST(I),
     1 PERCST(I),(WASTN(J,I),J=1,2),VOL(I),AVVOL(I),PERVOL(I),I=1,3)
```

```
      WRITE(6,505) (COSTN(J,4),J=1,2),COST(4),AVCOST(4),UNCOST(4),
     1             PERCST(4)
      WRITE(6,5050)((COSTN(J,I),J=1,2),COST(I),AVCOST(I),UNCOST(I),
     1             I=5,6)
      WRITE(6,506)
      NLINES=NLINES+11
      KMAX=MAX0(NUM(1),NUM(2),NUM(3),NUM(4))
      NR=1
      NMKT=NUMMKT(1)
      NM=0
      DO 1 K=1,81
      NM=NM+1
      IF (NM.LE.NMKT) GO TO 70
      NM=1
      NR=NR+1
      NMKT=NUMMKT(NR)
70    IF(K.GT.KMAX) GO TO 7
      TNAM=BLANK8
      TUSE=BLANK4
      PNAM=BLANK8
      PUSE=BLANK4
      LNAM=BLANK8
      LUSE=BLANK4
      MNAM=BLANK8
      MUSE=BLANK4
      IF(K.GT.NUM(1)) GO TO 3
      TNAM=TRFNAM(K)
      TUSE=NO
      IF(DT(K).GT.0.) TUSE=YES
3     IF(K.GT.NUM(2)) GO TO 4
      PNAM=PLTNAM(K)
      PUSE=NO
      IF(DP(K).GT.0.) PUSE=YES
4     IF(K.GT.NUM(3)) GO TO 5
      LNAM=LFLNAM(K)
      LUSE=NO
      IF(DL(K).GT.0.) LUSE=YES
5     IF(K.GT.NUM(4)) GO TO 6
      MNAM=MKTNAM(NR,NM)
      MUSE=NO
      IF(WM(K).GT.0.) MUSE=YES
6     WRITE(6,507) TNAM,TUSE,PNAM,PUSE,LNAM,LUSE,MNAM,MUSE
      IF (NLINES.LT.LINPAG) GO TO 30
      CALL HEADER(1,1)
      WRITE (6,506)
      NLINES=NLINES+5
 30   NLINES=NLINES+1
1     CONTINUE
      IF(NLINES.GE.LINPAG+8)  CALL HEADER(1,1)
7     WRITE(6,508)
      DO8 N=1,3
      NUMB=NUM(N)
      CAP(N)=0.
      OP(N) =0.
      TR(N)=0.
      DO9 J=1,NUMB
      CAP(N)=CAP(N)+FCP(J,N)*DELTA(J,N)+VCP(J,N)*WST(J,N)
      OP(N) =OP(N) +(FOP(J,N)*DELTA(J,N)+VOP(J,N)*WST(J,N))*DISC
 9    CONTINUE
      IF(N.EQ.3) GO TO 59
      IF(N.EQ.2) GO TO 40
C
C
C           TP OR TL ARC
C
C
      NARCLO=NANOSL+1
      NARCHI=NANOTL
      GO TO 50
C
C
C           PL OR PM ARC
C
C
 40   NARCLO=NANOTL
      NARCHI=NANOPM
 50   DO55 NARC=NARCLO,NARCHI
      TR(N)=TR(N)+ARCOST(NARC)*ARCWST(NARC)*DISC
 55   CONTINUE
 59   TOT(N)=CAP(N)+OP(N)+TR(N)
 8    PER(N)=TOT(N)/COST(4)*100.
      WRITE(6,509) (COSTN(I,1),I=1,2),BLANK8,BLANK8,(CAP(N),N=1,3)
      WRITE(6,509) (COSTN(I,2),I=1,2),BLANK8,BLANK8,(OP(N),N=1,3)
      WRITE(6,509) (COSTN(I,3),I=1,2),BLANK8,BLANK8,(TR(N),N=1,3)
      WRITE(6,509) (TOTOF(I),I=1,2),(FACIL(I),I=1,2),(TOT(N),N=1,3)
      WRITE(6,5090) (PERCNT(I),I=1,2),(SYSTOT(I),I=1,2),(PER(N),N=1,3)
504   FORMAT("0","SYSTEM FINANCIAL SUMMARY",T78,"MATERIAL FLOW IN THE",
     1 " SYSTEM"//T24,"STUDY",T37,"ANNUAL",T48,"UNIT",T57,"PERCENT",T99,
     2 "STUDY",T111,"ANNUAL",T123,"PERCENT"/T24,"PERIOD",T36,"AVERAGE",
     3 T57,"OF TOTAL",T99,"PERIOD",T110,"AVERAGE",T123,"OF TOTAL"/T25,
     4 "($)",T37,"($/YR)",T48,"($/T)",T58,"(%)",T100,"($)",T110,
     5 "($/YR)",T124,"(%)"/)
505   FORMAT(" ",2A8,2X,2F12.2,F12.5,F9.1,13X,2A8,2X,2F12.2,F9.1)
5050  FORMAT(" ",2A8,2X,2F12.2,F12.5)
506   FORMAT("0"," FACILITIES AND MARKETS USED IN THE SYSTEM"//T3,
     1 "TRANSFER STATIONS",T24,"USED",T38,"PROCESSING PLANTS",T59,
     2 "USED",T73,"LANDFILLS",T86,"USED",T100,"MARKETS",T113,"USED"/)
507   FORMAT(" ",T7,A8,T24,A4,T42,A8,T59,A4,T74,A8,T86,A4,T100,A8,T113,
     1            A4)
508   FORMAT("0","COSTS OF FACILITIES USED IN THE SYSTEM"//T38,"TRANS",
     1 "FER",9X,"PROCESSING",9X,"LANDFILL"//T40,"($)",T60,"($)",T76,
     2 "($)"/)
509   FORMAT(" ",4A8,3(F15.0,2X))
5090  FORMAT(" ",4A8,3(F15.1,2X))
      RETURN
      END
```

```
C
C
C
      SUBROUTINE SUMREP
C
C
C  ************************************************************
C  *                                                          *
C  *   THIS SUBROUTINE WRITES THE  SUMMARY REPORT OF WASTE    *
C  *   GENERATION AREA SHIPMENTS.                             *
C  *                                                          *
C  ************************************************************
C
C
C
       IMPLICIT REAL(A-H,L,O-Z),INTEGER(I-K,M,N)
C
C      ***    DECLARATION OF REAL VARIABLES (FOR TITLES AND
C      ***                    NAMES)
       REAL TITLE,RUNNAM,VECNAM,SRCNAM,
     1        NAM,RSCNAM,MKTNAM,DEST,SOURCE
C
       INTEGER T,D,R,W,LINPAG
      INTEGER  RUNBEG,RUNEND,INREP,OUTREP,
     1            VECIDN,SRCIDN,TRFIDN,PLTIDN,LFLIDN,MKTIDN,
     2            ORIGID,DESTID
      INTEGER TYP(3),TYPE,SRC(200)
C
C      *** GENERAL COMMON BLOCK
C
      COMMON/NUMBS/NPAGE,NLINES,LINPAG,
     1               RATE,NUMYRS,INREP,OUTREP,
     2               NUMVEC,NUMSRC,NUM(4),NRESRC,
     3               NUMMKT(9)
C
C      ***    VEHICLES COMMON BLOCK
C
       COMMON/VECDAT/VECNAM(8),COSTTH(8),VECIDN
C
C      ***    SOURCES COMMON BLOCK
C
       COMMON/SRCDAT/SRCNAM(200),SRCTOT(200),SRCIDN(200),ALWAST
C
C      ***    FACILITIES COMMON BLOCK
C
      COMMON/FACDAT/NAM(20,3),IDN(20,3),FCP(20,3),VCP(20,3),FOP(20,3),
     1              VOP(20,3)
C
C      ***    COMMON FOR MARKETS
       COMMON/MKTDAT/RSCNAM(9),MKTNAM(9,9),MKTIDN(9,9),DEMAND(9,9),
     1               PRICET(9,9)
C
C      ***    COMMON FOR TRANSPORTATION ARCS
C
       COMMON/ARCDAT/ARCOST(5000),ORIGID(5000),DESTID(5000),
     1        NANDST,NANDSP,NANDSL,NANDTP,NANDTL,NANDPL,NANDPM,NANDWL
C
```

```
      COMMON/OUT/ARCWST(5000),WST(20,3),WM(81),DELTA(20,3),COST(6),DISC
C
      DATA TYP/" TS "," PR "," LF "/
C
      CALL HEADER(2,0)
      WRITE(6,510)
      WRITE(6,512)
      NLINES=NLINES+12
      SRC(1)=ORIGID(1)
      KMAX=1
      DO1 N=1,NANDSL
      DO2 K=1,KMAX
 2    IF(ORIGID(N).EQ.SRC(K)) GO TO 1
      KMAX=K
      SRC(KMAX)=ORIGID(N)
 1    CONTINUE
      DO3 K=1,KMAX
      DO4 J=1,NUMSRC
 4    IF(SRCIDN(J).EQ.SRC(K)) SOURCE=SRCNAM(J)
      DO5 I=1,NANDSL
      IF(ARCWST(I).LE.0.) GO TO 5
      IF(ORIGID(I).NE.SRC(K)) GO TO 5
      IF(I.LE.NANDST) NTYP=1
      IF(I.GT.NANDST.AND.I.LE.NANDSP) NTYP=2
      IF(I.GT.NANDSP.AND.I.LE.NANDSL) NTYP=3
      TYPE=TYP(NTYP)
      NUMB=NUM(NTYP)
      DO10 J=1,NUMB
 10   IF(IDN(J,NTYP).EQ.DESTID(I)) GO TO 11
 11   DEST=NAM(J,NTYP)
      WASTE=ARCWST(I)
      AV=0.
      TRCOST=ARCWST(I)*ARCOST(I)*DISC
      IF(WASTE.NE.0.) AV=TRCOST/WASTE
      TRCOST=TRCOST/1000.
      YRCOST=TRCOST/NUMYRS
      YRWAST=WASTE/NUMYRS/1000.
      WASTE=WASTE/1000.
      IF (NLINES.LT.LINPAG) GO TO 20
      CALL HEADER (2,1)
      WRITE (6,510)
      WRITE (6,512)
      NLINES=NLINES+12
  20  WRITE(6,513) SOURCE,DEST,TYPE,WASTE,YRWAST,TRCOST,YRCOST,AV
      NLINES=NLINES+1
 5    CONTINUE
 3    CONTINUE
C
C
C                         FORMATS
C
C
 510  FORMAT("0",T11,"FACILITY TYPES:  TS-TRANSFER STATION,  PR-",
     1 "PROCESSING PLANT,  LF-LANDFILL"/)
 512  FORMAT("0",T6,"QUANTITIES ARE IN THOUSANDS OF TONNES.  TOTAL ",
     1 "COST AND YEARLY AVERAGE COST ARE"/T6,"IN THOUSANDS OF DOLLARS,",
     2 " AVERAGE IN DOLLARS PER TONNE."///T15,"-------------------",
    3 "------------------      SHIPPED TO FACILITY    ------------------",
    4 "----------------------"/T4,"GENERATION",T17,"NAME",T25,"TYPE",
    5 T35,"TOTAL",T49,"YEARLY",T65,"TOTAL",T79,"ANNUAL",T94,"AVERAGE"/,
    6 T8,"AREA",T35,"QUANTITY",T49,"AVERAGE",T65,"COST",T79,"AVERAGE",
    7 T94,"COST/TON"/)
513  FORMAT(4X,A8,T15,A8,T26,A4,T32,F10.1,T46,F10.1,T62,F10.1,T76,
    1  F10.1,T91,F10.5)
     RETURN
     END
```

```
      SUBROUTINE DETFAC(FACNAM,FACTYP)
C
C
C  **********************************************************
C  *                                                        *
C  *  THIS SUBROUTINE WRITES THE DETAILED REPORT OF FACILITY *
C  *  OPERATIONS FOR EITHER ONE FACILITY, SPECIFIED BY NAME  *
C  *  (FACNAM), OR ALL FACILITIES (FACNAM="ALL      ").      *
C  *                                                        *
C  **********************************************************
C
C
C
      REAL ALL,FACNAM,RSCNAM,MKTNAM,NAM,DESNAM
      INTEGER TYPE(3),FACTYP,ORIGID,DESTID,DESTYP,LINPAG
C
C       *** GENERAL COMMON BLOCK
C
      COMMON/NUMBS/NPAGE,NLINES,LINPAG,
     1              RATE,NUMYRS,INREP,OUTREP,
     2              NUMVEC,NUMSRC,NUM(4),NRESRC,
     3              NUMMKT(9)
C
C
C       ***   FACILITIES COMMON BLOCK
C
      COMMON/FACDAT/NAM(20,3),IDN(20,3),FCP(20,3),VCP(20,3),FOP(20,3),
     1              VOP(20,3)
C
C       ***   COMMON FOR MARKETS
       COMMON/MKTDAT/RSCNAM(9),MKTNAM(9,9),MKTIDN(9,9),DEMAND(9,9),
     1              PRICET(9,9)
C
C       ***   COMMON FOR TRANSPORTATION ARCS
C
       COMMON/ARCDAT/ARCOST(5000),ORIGID(5000),DESTID(5000),
     1         NANDST,NANDSP,NANDSL,NANDTP,NANDTL,NANDPL,NANDPM,NANDWL
C
      COMMON/OUT/ARCWST(5000),WST(20,3),WM(81),DELTA(20,3),COST(6),DISC
C
      DATA TYPE(1)/" TS "/,TYPE(2)/" PR "/,TYPE(3)/" LF "/
      DATA ALL/"ALL      "/
C
      CALL HEADER(4,0)
      WRITE(6,514) FACNAM
      WRITE(6,515)
      NLINES=NLINES+5
      DO1 NTYPE=1,3
      IF(FACNAM.EQ.ALL) GO TO 10
      IF(FACTYP.NE.TYPE(NTYPE)) GO TO 1
 10   NUMB=NUM(NTYPE)
      DO2 J=1,NUMB
      IF(FACNAM.EQ.ALL) GO TO 3
      IF(NAM(J,NTYPE).NE.FACNAM) GO TO 2
 3    TRCOST=0.
      UNITTR=0.
      UNITCP=0.
      UNITOP=0.
      UNITOT=0.
      IF(NTYPE.EQ.3) GO TO 8
      IF(NTYPE.EQ.2) GO TO 4
C
C              TP OR TL ARC
C
      NARCLO=NANDSL+1
      NARCHI=NANDTL
      GO TO 5
C
C              PL OR PM ARC
C
    4 NARCLO=NANDTL+1
      NARCHI=NANDPM
C
    5 CONTINUE
      DO6 N=NARCLO,NARCHI
    6 IF(ORIGID(N).EQ.IDN(J,NTYPE)) TRCOST=TRCOST+ARCOST(N)*ARCWST(N)
     1                                                    *DISC
      IF(WST(J,NTYPE).NE.0.) UNITTR=TRCOST/WST(J,NTYPE)
    8 CAPCOS=FCP(J,NTYPE)*DELTA(J,NTYPE)+VCP(J,NTYPE)*WST(J,NTYPE)
      OPCOST=(FOP(J,NTYPE)*DELTA(J,NTYPE)+VOP(J,NTYPE)*WST(J,NTYPE))
     1                         *DISC
      IF(WST(J,NTYPE).NE.0.) UNITCP=CAPCOS/WST(J,NTYPE)
      IF(WST(J,NTYPE).NE.0.) UNITOP=OPCOST/WST(J,NTYPE)
      TOTCOS=CAPCOS+OPCOST+TRCOST
      IF(WST(J,NTYPE).NE.0.) UNITOT=TOTCOS/WST(J,NTYPE)
      IF (NLINES.LT.LINPAG) GO TO 20
      CALL HEADER (4,1)
      WRITE (6,514) FACNAM
      WRITE (6,515)
      NLINES=NLINES+8
   20 NLINES=NLINES+1
      WRITE(6,516) NAM(J,NTYPE),CAPCOS,OPCOST,TRCOST,TOTCOS,UNITCP,
     1             UNITOP,UNITTR,UNITOT,WST(J,NTYPE)
      IF(FACNAM.NE.ALL) GO TO 80
    2 CONTINUE
    1 CONTINUE
      IF(FACNAM.EQ.ALL) RETURN
   80 IF(NTYPE.EQ.3) RETURN
      WRITE(6,517)
      NLINES=NLINES+6
      DO9 N=NARCLO,NARCHI
      IF(ORIGID(N).NE.IDN(J,NTYPE)) GO TO 94
      IF(N.GT.NANDSL.AND.N.LE.NANDTP) DESTYP=2
      IF(N.GT.NANDTP.AND.N.LE.NANDTL) DESTYP=3
      IF(N.GT.NANDTL.AND.N.LE.NANDPL) DESTYP=3
      IF(N.GT.NANDPL.AND.N.LE.NANDPM) DESTYP=4
      VOLWST=ARCWST(N)
   94 NUMB=NUM(DESTYP)
      IF(DESTYP.EQ.4) GO TO 100
      DO95 M=1,NUMB
   95 IF(DESTID(N).EQ.IDN(M,DESTYP)) GO TO 96
   96 DESNAM=NAM(M,DESTYP)
      GO TO 110
  100 DO101 I=1,NRESRC
```

```
      NMKT=NUMMKT(I)
      DO101 K=1,NMKT
 101  IF(DESTIO(N).EQ.MKTION(I,K)) GO TO 102
 102  DESNAM=MKTNAM(I,K)
      IF (NLINES.LT.LINPAG) GO TO 30
      CALL HEADER (4,1)
      WRITE (6,517)
      NLINES=NLINES+5
  30  NLINES=NLINES+1
 110  WRITE(6,518) DESNAM,VOLWST
  9   CONTINUE
C
C
C                         FORMATS
C
C
 514  FORMAT("0","FACILITIES INCLUDED IN THIS REPORT:  ",A8)
 515  FORMAT("0",T15,"-------------------  ANNUAL COSTS  -",
     1 "-----------------",4X,"-------------",
     2 "----  UNIT COSTS  ---------------"//T4,"FACILITY",
     3 T17,"CAPITAL",T29,"OPERATING",T43,"TRANSPORT",T56,"TOTAL",T73,
     4 "CAPITAL",T85,"OPERATING",T99,"TRANSPORT",T112,"TOTAL",T124,
     5 "WASTE"/T18,"COST",T33,"COST",T47,"COST",T56,"COST",T74,"COST",
     6 T89,"COST",T103,"COST",T112,"COST",T123,"SHIPPED"/T19,"($)",T34,
     7 "($)",T48,"($)",T57,"($)",T74,"($/T)",T89,"($/T)",T103,"($/T)",
     8 T112,"($/T)",T125,"(T)"//)
 516  FORMAT(" ",3X,A8,2X,4(2X,F10.0,1X),4(2X,F10.5,1X),3X,F10.0)
 517  FORMAT("-",10X,"QUANTITY SHIPPED TO EACH DESTINATION"//T15,
     1 "DESTINATION",T30,"AMOUNT (IN TONNES)"/)
 518  FORMAT(" ",T17,A8,T33,F10.0)
C
C
      RETURN
      END
      SUBROUTINE DETGEN(GENNAM)
C
C
C  *****************************************************************
C  *                                                               *
C  *  THIS SUBROUTINE WRITES THE DETAILED REPORT OF WASTE GENERATION *
C  *  AREA SHIPMENTS FOR EITHER ONE GENERATION AREA, SPECIFIED BY    *
C  *  NAME (GENNAM), OR ALL THE GENERATION AREAS (GENNAM="ALL      "). *
C  *                                                               *
C  *****************************************************************
C
C
      INTEGER SRCIDN,MKTIDN,ORIGID,DESTID,DESTYP,LINPAG
C
      REAL ALL,GENNAM,SRCNAM,RSCNAM,MKTNAM,NAM
C
C       *** GENERAL COMMON BLOCK
C
      COMMON/NUMBS/NPAGE,NLINES,LINPAG,
     1              RATE,NUMYRS,INREP,OUTREP,
     2              NUMVEC,NUMSRC,NUM(4),NRESRC,
     3              NUMMKT(9)
C
C
C       ***   FACILITIES COMMON BLOCK
C
      COMMON/FACDAT/NAM(20,3),IDN(20,3),FCP(20,3),VCP(20,3),FOP(20,3),
     1             VOP(20,3)
C
C       ***   SOURCES COMMON BLOCK
C
       COMMON/SRCDAT/SRCNAM(200),SRCTOT(200),SRCIDN(200),ALWAST
C
C
C       ***   COMMON FOR MARKETS
       COMMON/MKTDAT/RSCNAM(9),MKTNAM(9,9),MKTIDN(9,9),DEMAND(9,9),
     1              PRICET(9,9)
C
C       ***   COMMON FOR TRANSPORTATION ARCS
C
       COMMON/ARCDAT/ARCOST(5000),ORIGID(5000),DESTID(5000),
     1        NANDST,NANDSP,NANDSL,NANDTP,NANDTL,NANDPL,NANDPM,NANDHL
C
      COMMON/OUT/ARCWST(5000),WST(20,3),WM(81),DELTA(20,3),COST(6),DISC
      COMMON/GEN/NOSUBT,IDSUBT(40)
C
      DATA ALL/"ALL      "/
C
C
      CALL HEADER(3,0)
      WRITE(6,519) GENNAM
      WRITE(6,520)
      NLINES=NLINES+8
      DO1 J=1,NUMSRC
      IF(GENNAM.EQ.SRCNAM(J)) GO TO 2
      IF(GENNAM.NE.ALL) GO TO 1
      GO TO 3
```

```
 2    JJ=J
 3    WSTGEN=0.
      TRCOST=0.
      DO4 N=1,NANDSL
      IF(ORIGID(N).NE.SRCIDN(J)) GO TO 4
      WSTGEN=WSTGEN+ARCWST(N)
      TRCOST=TRCOST+ARCOST(N)*ARCWST(N)*DISC
 4    CONTINUE
      UNITTR=TRCOST/WSTGEN
      TRAVG=TRCOST/NUMYRS
      WSTAVG=WSTGEN/NUMYRS
      IF (NLINES.LT.LINPAG) GO TO 20
      CALL HEADER (3,1)
      WRITE (6,519) GENNAM
      WRITE (6,520)
      NLINES=NLINES+8
 20   WRITE(6,521) SRCNAM(J),TRCOST,TRAVG,UNITTR,WSTGEN,WSTAVG
      NLINES=NLINES+1
 1    CONTINUE
      IF(GENNAM.NE.ALL) GO TO 90
      IF(NOSUBT.EQ.0) RETURN
      IF(NLINES.GT.LINPAG) CALL HEADER(3,1)
      WRITE(6,524)
      NLINES=NLINES+8
      DO8 J=1,NOSUBT
        IF(J.EQ.1)IDLO=0
           IF(J.GT.1)IDLO=IDSUBT(J-1)
         IDHI=IDSUBT(J)
      WSTGEN=0.
      TRCOST=0.
      DO9 N=1,NANDSL
      IF(ORIGID(N).LE.IDLO.OR.ORIGID(N).GT.IDHI)GO TO 9
      WSTGEN=WSTGEN+ARCWST(N)
      TRCOST=TRCOST+ARCOST(N)*ARCWST(N)*DISC
 9    CONTINUE
      UNITTR=TRCOST/WSTGEN
      TRAVG=TRCOST/NUMYRS
      WSTAVG=WSTGEN/NUMYRS
      IF (NLINES.LT.LINPAG) GO TO 30
      CALL HEADER (3,1)
      WRITE (6,524)
      NLINES=NLINES+8
 30   WRITE(6,525) IDSUBT(J),TRCOST,TRAVG,UNITTR,WSTGEN,WSTAVG
      NLINES=NLINES+1
 8    CONTINUE
      RETURN
 90   WRITE(6,522)
      NLINES=NLINES+6
      DO5 N=1,NANDSL
      IF(ORIGID(N).NE.SRCIDN(JJ)) GO TO 5
      IF(N.LT.NANDST) DESTYP=1
      IF(N.GT.NANDST.AND.N.LE.NANDSP) DESTYP=2
      IF(N.GT.NANDSP.AND.N.LE.NANDSL) DESTYP=3
      NUMB=NUM(DESTYP)
      DO6 M=1,NUMB
 6    IF(DESTID(N).EQ.IDN(M,DESTYP)) GO TO 7
 7    IF(NLINES.GE.LINPAG) CALL HEADER(3,1)
```

```
      NLINES=NLINES+1
      WRITE(6,523) NAM(M,DESTYP),ARCHST(N)
 5    CONTINUE
C
C
C                         FORMATS
C
C
 519  FORMAT("0","GENERATION AREA(S) INCLUDED IN THIS REPORT:  ",A8)
 520  FORMAT("0",T5,"GENERATION",T21,"TRANSPORT",T38,"AVERAGE",T54,
     1  "UNIT",T67,"QUANTITY",T82,"AVERAGE"/T6,"AREA NAME",T25,"COST",
     2  T37,"TRANSPORT",T52,"TRANSPORT",T67,"OF WASTE",T83,"WASTE"/T39,
     3  "COST/YR.",T53,"COST",T66,"GENERATED",T82,"GEN./YR."/T25,"($)",
     4  T41,"($)",T53,"($)",T71,"(T)",T86,"(T)"/)
 521  FORMAT(" ",T6,A8,2F15.0,F15.5,2F15.0)
 522  FORMAT("-",T5,"---  QUANTITY SHIPPED TO EACH DESTINATION  ---",
     1  //T10,"DESTINATION",T24,"AMOUNT IN TONNES"/)
 523  FORMAT(" ",T11,A8,T27,F10.0)
 524  FORMAT("0",T5,"SUBTOTAL TO",T21,"TRANSPORT",T38,"AVERAGE",T54,
     1  "UNIT",T67,"QUANTITY",T82,"AVERAGE"/T4,"SOURCE ID NO.",T25,
     2  "COST",T37,"TRANSPORT",T52,"TRANSPORT",T67,"OF WASTE",T83,
     3  "WASTE"/T39,"COST/YR.",T53,"COST",T66,"GENERATED",T82,"GEN.",
     4  "/YR."/T25,"($)",T41,"($)",T53,"($)",T71,"(T)",T86,"(T)"/)
 525  FORMAT(" ",T9,I3,T14,2F15.0,F15.5,2F15.0)
      RETURN
      END
      SUBROUTINE MKTREP
C
C    ***********************************************
C    *                                             *
C    *  THIS SUBROUTINE WRITES THE MARKETING REPORT.  *
C    *                                             *
C    ***********************************************
C
C
C
C       ***   INITIAL DECLARATION OF L VARIABLES AS REAL,
C       ***   ALL IMPLICIT INTEGER VARIABLES AS INTEGER
C
       IMPLICIT REAL(A-H,L,O-Z),INTEGER(I-K,M,N)
C
C       ***   DECLARATION OF REAL VARIABLES (FOR TITLES AND
C       ***                          NAMES)
       REAL TITLE,RUNNAM,VECNAM,SRCNAM,
     1          RSCNAM,MKTNAM
C
       INTEGER T,O,R,W,LINPAG
      INTEGER  RUNBEG,RUNEND,INREP,OUTREP,
     1            VECION,SRCION,TRFION,PLTION,LFLION,MKTION,
     2            ORIGID,DESTID
      COMMON/NUMBS/NPAGE,NLINES,LINPAG,
     1                 RATE,NUMYRS,INREP,OUTREP,
     2                 NUMVEC,NUMSRC,NUMTRF,NUMPLT,NUMLFL,NRM,NRESRC,
     3                 NUMMKT(9)
       COMMON/ARCDAT/ARCOST(5000),ORIGID(5000),DESTID(5000),
     1          NANDST,NANDSP,NANDSL,NANDTP,NANDTL,NANDPL,NANDPM,NANDWL
      COMMON/OUT/ARCWST(5000),WST(20,3),WM(81),DELTA(20,3),COST(6),DISC
C       ***   COMMON FOR MARKETS
       COMMON/MKTDAT/RSCNAM(9),MKTNAM(9,9),MKTION(9,9),DEMAND(9,9),
     1                 PRICET(9,9)
C
C
C
C            IF THERE ARE NO RESOURCES, NO REPORT IS WRITTEN.
C
      IF(NRESRC.EQ.0) RETURN
C
C
      CALL HEADER(5,0)
      WRITE(6,530)
      NLINES=NLINES+5
      NAND=NANDWL
      TOTWST=0.
      TOTREV=0.
      DO1 NR=1,NRESRC
      NMKT=NUMMKT(NR)
      DO2  M=1,NMKT
      IF(NR.EQ.1) K=M
      IF(NR.GT.1) K=NUMMKT(NR-1)+M
      NAND=NAND+1
      IF(WM(K).LE.0.) GO TO 2
      PRICET(NR,M)=-PRICET(NR,M)*DISC
      REV=WM(K)*PRICET(NR,M)
      IF (NLINES.LT.LINPAG) GO TO 20
      CALL HEADER (5,1)
      WRITE (6,530)
      NLINES=NLINES+5
 20   NLINES=NLINES+1
      WRITE(6,531) MKTNAM(NR,M),RSCNAM(NR),WM(K),PRICET(NR,M),REV
      TOTWST=TOTWST+WM(K)
      TOTREV=TOTREV+REV
2     CONTINUE
1     CONTINUE
      IF(NLINES.GE.LINPAG) CALL HEADER(5,1)
      WRITE(6,532) TOTWST,TOTREV
530   FORMAT("-",T5,"MARKET NAME",T20,"PRODUCT NAME",T37,"QUANTITY",T54,
     1 "MARKET PRICE",T71,"REVENUE"/T40,
     2 "(T)",T59,"($)",T74,"($)"/)
531   FORMAT(" ",T7,A8,T22,A8,T35,3(F11.0,6X))
532   FORMAT("0",T22,"TOTALS",T35,F11.0,23X,F11.0)
      RETURN
      END
```

```
C                                                                        1003.
      SUBROUTINE TPSHIP                                                   541.
C                                                                         542.
C                                                                         543.
C     *******************************************************            544.
C     *                                                     *            545.
C     *  THIS SUBROUTINE WRITES THE TRANSFER STATION SHIPMENT *          546.
C     *  REPORT                                               *          547.
C     *                                                     *            548.
C     *******************************************************            549.
C                                                                         550.
C                                                                         551.
C                                                                         552.
       IMPLICIT REAL(L)                                                   553.
C                                                                         554.
C      ***   DECLARATION OF REAL VARIABLES (FOR TITLES AND                555.
C      ***                             NAMES)                             556.
       REAL TITLE,RUNNAM,VECNAM,SRCNAM,                                   557.
     1          NAM,RSCNAM,MKTNAM,DEST,SOURCE                             558.
C                                                                         559.
       INTEGER T,D,R,W,F,LINPAG,RUNYRS                                    560.
      INTEGER  RUNBEG,RUNEND,INREP,OUTREP,                                561.
     1             VECIDN,SRCIDN,TRFIDN,PLTIDN,LFLIDN,MKTIDN,             562.
     2             ORIGID,DESTID                                          563.
      INTEGER TYP(4),TYPE,SRC(200)                                        564.
C                                                                         565.
C      *** GENERAL COMMON BLOCK                                           566.
C                                                                         567.
      COMMON /HEAD/ TITLE(5),RUNNAM(5),RUNBEG,RUNEND,DAT
      COMMON/NUMBS/NPAGE,NLINES,LINPAG,                                   568.
     1              RATE,NUMYRS,INREP,OUTREP,                             569.
     2              NUMVEC,NUMSRC,NUMTRF,NUMPLT,NUMLFL,NRM,NRESRC,        570.
     3              NUMMKT(9)                                             571.
C                                                                         572.
C      ***   VEHICLES COMMON BLOCK                                        573.
C                                                                         574.
       COMMON/VECDAT/VECNAM(8),COSTTH(8),VECIDN                           575.
C                                                                         576.
C      ***   SOURCES COMMON BLOCK                                         577.
C                                                                         578.
       COMMON/SRCDAT/SRCNAM(200),SRCTOT(200),SRCIDN(200),ALWAST           579.
C                                                                         580.
C      ***   FACILITIES COMMON BLOCK                                      581.
C                                                                         582.
      COMMON/FACDAT/TRFNAM(20),PLTNAM(20),LFLNAM(20),TRFIDN(20),          583.
     1 PLTIDN(20),LFLIDN(20),TRFFCP(20),PLTFCP(20),LFLFCP(20),            584.
     2 TRFVCP(20),PLTVCP(20),LFLVCP(20),TRFFOP(20),PLTFOP(20),
     3 LFLFOP(20),TRFVOP(20),PLTVOP(20),LFLVOP(20)
C                                                                         585.
C      ***   COMMON FOR MARKETS                                           586.
       COMMON/MKTDAT/RSCNAM(9),MKTNAM(9,9),MKTIDN(9,9),DEMAND(9,9),       587.
     1              PRICET(9,9)                                           588.
C                                                                         589.
C      ***   COMMON FOR TRANSPORTATION ARCS                               590.
C                                                                         591.
       COMMON/ARCDAT/ARCOST(5000),ORIGID(5000),DESTID(5000),              592.
     1          NANDST,NANDSP,NANDSL,NANDTP,NANDTL,NANDPL,NANDPM,NANDWL   593.
```

```
C
      COMMON/OUT/ARCWST(5000),WST(20,3),WM(81),DELTA(20,3),COST(6),DISC
C
      DATA TYP/" PL "," LF "," MK "," TR "/
C
      AV=0.
      CALL HEADER(6,0)
      WRITE(6,510)
      WRITE(6,512)
      NLINES=NLINES+11
      IF (NUMTRF.EQ.0) GO TO 101
C
C       *** SHIPMENTS FROM TRANSFER STATIONS ARE ON TP OF TL ARCS
C
      NNDSL1=NANDSL+1
      DO 100 I=1,NUMTRF
      SOURCE=TRFNAM(I)
      DO 50 J=NNDSL1,NANDTL
      IF (ORIGID(J).NE.TRFIDN(I)) GO TO 50
      IF (ARCWST(J).LE.0) GO TO 50
      IF (J.LE.NANDTP) NTYP=1
      IF (J.GT.NANDTP) NTYP=2
      TYPE=TYP(NTYP)
      IF (NTYP.EQ.2) GO TO 15
      DO 10 K=1,NUMPLT
 10   IF (DESTID(J).EQ.PLTIDN(K)) DEST=PLTNAM(K)
      GO TO 30
 15   DO 20 K=1,NUMLFL
 20   IF (DESTID(J).EQ.LFLIDN(K)) DEST=LFLNAM(K)
 30   CONTINUE
      WASTE=ARCWST(J)
      TRCOST=ARCWST(J)*ARCOST(J)*DISC
      IF (WASTE.NE.0) AV=TRCOST/WASTE
      TRCOST=TRCOST/1000.
      YRCOST=TRCOST/NUMYRS
      YRWAST=WASTE/NUMYRS/1000.
      WASTE=WASTE/1000.
      IF (NLINES.LT.LINPAG) GO TO 40
      CALL HEADER (6,1)
      WRITE (6,510)
      WRITE (6,512)
      NLINES=NLINES+11
  40  WRITE (6,513) SOURCE,TYP(4),DEST,TYPE,WASTE,YRWAST,TRCOST,
     1 YRCOST,AV
      NLINES=NLINES+1
 50   CONTINUE
 100  CONTINUE
C
 101  IF (NUMPLT.EQ.0) RETURN
C
C         **** SHIPMENTS FROM PLANTS ARE ON PL OR PM ARCS
C
      NNDTL1=NANDTL+1
      DO 200 I=1,NUMPLT
      SOURCE=PLTNAM(I)
      DO 150 J=NNDTL1,NANDPM
      IF (ORIGID(J).NE.PLTIDN(I)) GO TO 150
      IF (ARCWST(J).LE.0) GO TO 150
      IF (J.LE.NANDPL) NTYP=2
      IF (J.GT.NANDPL) NTYP=3
      TYPE=TYP(NTYP)
      IF (NTYP.EQ.3) GO TO 115
      DO 110 K=1,NUMLFL
  110 IF (DESTID(J).EQ.LFLION(K)) DEST=LFLNAM(K)
      GO TO 130
  115 DO 120 K=1,NRESRC
      NMMKTK=NUMMKT(K)
      DO 120 KK=1,NMMKTK
  120 IF (DESTID(J).EQ.MKTION(K,KK)) DEST=MKTNAM(K,KK)
  130 CONTINUE
      WASTE=ARCWST(J)
      AV=0.
      TRCOST=ARCWST(J)*ARCOST(J)*DISC
      IF(WASTE.NE.0.) AV=TRCOST/WASTE
      TRCOST=TRCOST/1000.
      YRCOST=TRCOST/NUMYRS
      YRWAST=WASTE/NUMYRS/1000.
      WASTE=WASTE/1000.
      IF (NLINES.LT.LINPAG) GO TO 60
      CALL HEADER (6,1)
      WRITE (6,510)
      WRITE (6,512)
      NLINES=NLINES+11
   60 WRITE(6,513) SOURCE,TYP(1),DEST,TYPE,WASTE,YRWAST,TRCOST,YRCOST,
     1 AV
      NLINES=NLINES+1
  150 CONTINUE
  200 CONTINUE
C
C
C                             FORMATS
C
C
  510 FORMAT("0",T11,"FACILITY TYPES:  PL -- PROCESSING PLANT",
     1 ",  LF -- LANDFILL,   MK -- MARKET,  TR -- TRANSFER STN."/)
  512 FORMAT("0",T6,"QUANTITIES ARE IN THOUSANDS OF TONNES.  TOTAL ",
     1 "COST AND YEARLY AVERAGE COST ARE"/T6,"IN THOUSANDS OF DOLLARS,",
     2 " AVERAGE IN DOLLARS PER TONNE."///T5,"---- FROM ----",
     3T22,"------------",
     4 "------------------    SHIPPED TO FACILITY    ------------------",
     5 "-------------"/T5,"FACILITY  TYPE",T23,"NAME",T31,
     6 "TYPE",T41,"TOTAL",T55,"YEARLY",T71,"TOTAL",T85,"ANNUAL",T100,
     7"AVERAGE"/T41,"QUANTITY",T55,"AVERAGE",T71,"COST",T85,"AVERAGE",
     8 T99,"COST/TON"/)
  513 FORMAT(4X,A8,T15,A4,T21,A8,T31,A4,T38,F10.1,T51,F10.1,T66,
     1  F10.1,T82,F10.1,T96,F10.5)
       RETURN
       END
```

APPENDIX 9

SOURCE CODE OF TFETCH FOR IBM

```
C
      SUBROUTINE TFETCH
C
C
C  *****************************************************************
C  *                                                               *
C  *  THIS SUBROUTINE FETCHES THE SOLUTION FOUND                   *
C  *   BY THE MILP PACKAGE.                                        *
C  *                                                               *
C  *****************************************************************
C
C
      INTEGER NOLCOL,NOCOL2,I,J,L,M,N,P
      DOUBLE PRECISION NAME,ENDATA,ENDSEC,ROWS,COLUMS,RSECT,CSECT,
     1       ANAME,RNAM(6),ONTWST,ACTIV,SOLUTN
      DOUBLE PRECISION COLUMN(30),VALUES(25)
      INTEGER TYPE(200),VALNUM(50),ORIGID,DESTID
      REAL VALALF(50)
       DOUBLE PRECISION TITLE, RUNNAM
       INTEGER D,F,R,W,LINPAG,
     1         RUNBEG,RUNEND,RUNYRS,OUTREP
C
C     ***    GENERAL COMMON BLOCKS
C
      COMMON/GENRAL/TITLE(5),RUNNAM(5),D,F,R,W,LINPAG,
     1               RUNBEG,RUNEND,RUNYRS,OUTREP,DATE,DEDISC
      COMMON/NUMBS/NPAGE,NLINES,
     1               RATE,
     2               NUMVEC,NUMSRC,NUMTRF,NUMPLT,NUMLFL,NRM,NRESRC,
     3               NUMMKT(9)
       COMMON/ARCDAT/ARCOST(1000),ORIGID(1000),DESTID(1000),
     1         NANDST,NANDSP,NANDSL,NANDTP,NANDTL,NANDPL,NANDPM,NANDWL
      COMMON/OUT/ARCWST(1000),WST(20,3),WM(81),DELTA(20,3),COST(6),DISC
      EQUIVALENCE (VALUES(1),VALNUM(1),VALALF(1))
      DATA ENDATA/'ENDATA  '/,ENDSEC/'$ENDSEC$'/,ROWS/'ROWS     '/,
     1  COLUMS/'COLUMNS '/,RSECT/'RSECTION'/,CSECT/'CSECTIOND/,
     2  ANAME/'NAME    '/,ACTIV/'ACTIVITY'/,
     6  RNAM(1)/'CAPITALC'/,SOLUTN/'SOLUTION'/,
     7  RNAM(2)/'OPERATNG'/,RNAM(3)/'TRANSPRT'/,RNAM(5)/'REVENUES'/,
     8  RNAM(6)/'NETCOSTS'/,RNAM(4)/'TOTCOSTS'/,ONTWST/'ONTWASTE'/
      NANDWT=NANDPM+NUMTRF
      NANDWP=NANDWT+NUMPLT
      NANDWL=NANDWP+NUMLFL
      NRM=0
      IF(NRESRC.EQ.0) GO TO 101
      DO100 NR=1,NRESRC
  100 NRM=NRM+NUMMKT(NR)
  101 NANDWM=NANDWL+NRM
      NANDDT=NANDWM+NUMTRF
      NANDDP=NANDDT+NUMPLT
      NANDDL=NANDDP+NUMLFL
C
C         SKIP THE NAME,XDATA RECORD
C
 90   READ(F,END=500)
```

```
C
      READ(F)NAME,NOCOL
      CALL EXIT
      NOCOL2=2*NOCOL
      READ(F)(COLUMN(N),N=1 , NOCOL)
      READ(F)(TYPE(N),N=1,NOCOL2)
      J=0
      DO 2 I=2,NOCOL2,2
      J=J+TYPE(I)
2     CONTINUE
      J=J/4
      READ(F)(VALALF(N),N=1,J)
C
C         PRINT THE IDENTIFICATION ARRAY, ONE VALUE PER LINE
C
      WRITE(W,9)NAME
 9    FORMAT(1H1,35X,'PRINTOUT OF THE FILE ',A8)
C
      J=0
      DO 20 N=1,NOCOL
      L=J/4+1
      M=L+1
      P=L+19
      IF(TYPE(2*N-1) - 2)10,19,19
C
C         NUMERIC - INTEGER - VALUE
C
 10   IF(COLUMN(N).EQ.ROWS) NROWS=VALNUM(L)
      IF(COLUMN(N).EQ.ROWS) WRITE(W,11) COLUMN(N),VALNUM(L)
      IF(COLUMN(N).EQ.COLUMS) NCOLS=VALNUM(L)
      IF(COLUMN(N).EQ.COLUMS) WRITE(W,11) COLUMN(N),VALNUM(L)
11    FORMAT(1H0,35X,A8,' = ',I8)
19    J=J+TYPE(2*N)
20    CONTINUE
C
C         SKIP THE $ENDSEC$ OF THE IDENTIFICATION ARRAY.
C
      READ(F)
C
C         GET THE ROW AND COLUMN SECTIONS
C     ***************************************
C
21    READ(F)NAME,NOCOL
      IF(NAME.EQ.ENDATA)GO TO 90
22    WRITE(W,9)NAME
      READ(F)(COLUMN(N),N=1,NOCOL)
      READ(F)
      DO220 N=1,NOCOL
      IF(COLUMN(N).EQ.ACTIV) NACT=N
 220  IF(COLUMN(N).EQ.ANAME) NNAME=N
      WRITE(W,23) COLUMN(NACT),COLUMN(NNAME)
 23   FORMAT(1H0/1H ,A8,12X,A8)
      MINROW=NROWS-8
      MAXROW=MINROW+4
      IF(NAME.EQ.RSECT) COST(4)=0.
      I=0
 24    I=I+1
       READ(F) (VALUES(N),N=1,NOCOL)
       IF(VALUES(1).EQ.ENDSEC) GO TO 21
       IF(NAME.EQ.CSECT) GO TO 28
       IF(NAME.EQ.RSECT) GO TO 27
       WRITE(W,900) NAME
  900  FORMAT(' ','FILE ',A8,' FOUND WHEN RSECTION WAS EXPECTED.')
       CALL EXIT
 27    IF(I.LT.MINROW.OR.I.GT.MAXROW) GO TO 24
       DO240 J=1,5
       IF(J.LE.3) JP=J
       IF(J.GT.3) JP=J+1
       IF(VALUES(NNAME).EQ.RNAM(JP)) GO TO 241
 240   CONTINUE
 241   COST(JP)=VALUES(NACT)
       IF(JP.LE.3) COST(4)=COST(4)+COST(JP)
       IF(JP.EQ.5) WRITE(W,26) RNAM(4),COST(4),COST(4)
       WRITE(W,26) VALUES(NNAME),VALUES(NACT),COST(JP)
 26    FORMAT(1X,A12,2D20.8)
       GO TO 24
 28     CONTINUE
       IF(I.GT.NANDPM) GO TO 35
       ARCWST(I)=VALUES(NACT)
       GO TO 242
 35    IF(I.GT.NANDWT) GO TO 36
       NDIF=I-NANDPM
       WST(NDIF,1)=VALUES(NACT)
       GO TO 242
 36    IF(I.GT.NANDWP) GO TO 37
       NDIF=I-NANDWT
       WST(NDIF,2)=VALUES(NACT)
       GO TO 242
 37    IF(I.GT.NANDWL) GO TO 380
       NDIF=I-NANDWP
       WST(NDIF,3)=VALUES(NACT)
       GO TO 242
 380   IF(I.GT.NANDWM) GO TO 38
       NDIF=I-NANDWL
       WM(NDIF)=VALUES(NACT)
       GO TO 242
 38    IF(I.GT.NANDDT) GO TO 39
       NDIF=I-NANDWM
       DELTA(NDIF,1)=VALUES(NACT)
       GO TO 242
 39    IF(I.GT.NANDDP) GO TO 40
       NDIF=I-NANDDT
       DELTA(NDIF,2)=VALUES(NACT)
       GO TO 242
 40    IF(I.GT.NANDDL) GO TO 500
       NDIF=I-NANDDP
       DELTA(NDIF,3)=VALUES(NACT)
 242   WRITE(W,26) VALUES(NNAME),VALUES(NACT)
       GO TO 24
C
C       ALL T DATA NOW IN VECTORS.
C
500    RETURN
       END
```

TD
897
.J46
O58
1980